(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number : **0 561 685 A3**

(12) EUROPEAN PATENT APPLICATION

(21) Application number : **93400658.6**

(51) Int. Cl.^{5} : **G06F 12/14**

(22) Date of filing : **16.03.93**

(30) Priority : **16.03.92 JP 58048/92**

(43) Date of publication of application :
22.09.93 Bulletin 93/38

(84) Designated Contracting States :
DE FR GB

(88) Date of deferred publication of search report :
22.12.93 Bulletin 93/51

(71) Applicant : **FUJITSU LIMITED**
1015, Kamikodanaka Nakahara-ku
Kawasaki-shi Kanagawa 211 (JP)

(72) Inventor : **Hasebe, Takayuki, c/o FUJITSU LIMITED**
1015, Kamikodanaka, Nakahara-ku
Kawasaki-shi, Kanagawa 211 (JP)
Inventor : **Akiyama, Ryota, c/o FUJITSU LIMITED**
1015, Kamikodanaka, Nakahara-ku
Kawasaki-shi, Kanagawa 211 (JP)
Inventor : **Yoshioka, Makoto, c/o FUJITSU LIMITED**
1015, Kamikodanaka, Nakahara-ku
Kawasaki-shi, Kanagawa 211 (JP)

(74) Representative : **Joly, Jean-Jacques et al**
Cabinet Beau de Loménie 158, rue de l'Université
F-75340 Paris Cédex 07 (FR)

(54) An electronic data protection system.

(57) The system includes : a storage medium (1) for storing an encrypted electronic data, a medium number and encrypted permission information ; a vendor computer having a personal key generating unit (21) for generating a medium key based on the medium number (12), an electronic data decrypting key (22), and an encrypting unit (23) for encrypting the electronic data decrypting key based on the medium key to generate the encrypted permission information. The user computer having a personal key generating unit (31) for generating a medium key based on the medium number, a decrypting unit (32) for decrypting the encrypted permission information based on the medium key to generate the electronic data decrypting key which is the same as the electronic data decrypting key of the vendor computer, and a decrypting unit (34) for decrypting the encrypted electronic data (14) based on the electronic data decrypting key (33) to generate a plain text electronic data.

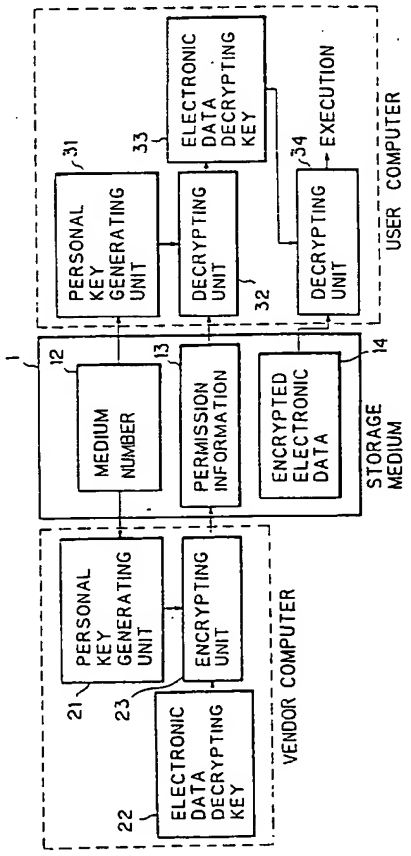


Fig. 2

EP 0 561 685 A3

European Patent Office

EUROPEAN SEARCH REPORT –

Application Number
EP 93 40 0658

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.5)
X Y A	WO-A-88 02202 (M/A-COM GOVERNMENT SYSTEMS INC.) * abstract; figures 2,3,6 * * page 3, line 1 - page 5, line 19 * * page 9, line 9 - line 13 * * page 11, line 15 - page 14, line 13 *	1,3,4,8 2,5-7,9 11,12	G06F12/14 G06F1/00
Y A	COMPUTER, vol. 17, no. 4 , April 1984 , LONG BEACH, US; pages 68 - 73 D.J.ALBERT ET AL 'Combatting Software Piracy by Encryption and Key Management' * the whole document *	2 1	
Y	EP-A-0 114 522 (SYNERTEK) * abstract *	5,6	
Y A	EP-A-0 268 139 (IBM) * abstract; figures 19,20 * * column 5, line 25 - column 10, line 27 * * column 19, line 9 - column 20, line 40 * * column 37, line 11 - line 57 *	7,9 11,12	TECHNICAL FIELDS SEARCHED (Int.Cl.5) G06F H04L
A	US-A-4 850 017 (IBM)		

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	28 October 1993	POWELL, D

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number : **0 561 685 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number : **93400658.6**

(51) Int. Cl.[5] : **G06F 12/14**

(22) Date of filing : **16.03.93**

(30) Priority : **16.03.92 JP 58048/92**

(43) Date of publication of application : **22.09.93 Bulletin 93/38**

(84) Designated Contracting States : **DE FR GB**

(71) Applicant : **FUJITSU LIMITED**
1015, Kamikodanaka Nakahara-ku
Kawasaki-shi Kanagawa 211 (JP)

(72) Inventor : **Hasebe, Takayuki, c/o FUJITSU LIMITED**
1015, Kamikodanaka, Nakahara-ku
Kawasaki-shi, Kanagawa 211 (JP)
Inventor : **Akiyama, Ryota, c/o FUJITSU LIMITED**
1015, Kamikodanaka, Nakahara-ku
Kawasaki-shi, Kanagawa 211 (JP)
Inventor : **Yoshioka, Makoto, c/o FUJITSU LIMITED**
1015, Kamikodanaka, Nakahara-ku
Kawasaki-shi, Kanagawa 211 (JP)

(74) Representative : **Joly, Jean-Jacques et al**
Cabinet Beau de Loménie 158, rue de l'Université
F-75340 Paris Cédex 07 (FR)

(54) An electronic data protection system.

(57) The system includes : a storage medium (1) for storing an encrypted electronic data, a medium number and encrypted permission information ; a vendor computer having a personal key generating unit (21) for generating a medium key based on the medium number (12), an electronic data decrypting key (22), and an encrypting unit (23) for encrypting the electronic data decrypting key based on the medium key to generate the encrypted permission information. The user computer having a personal key generating unit (31) for generating a medium key based on the medium number, a decrypting unit (32) for decrypting the encrypted permission information based on the medium key to generate the electronic data decrypting key which is the same as the electronic data decrypting key of the vendor computer, and a decrypting unit (34) for decrypting the encrypted electronic data (14) based on the electronic data decrypting key (33) to generate a plain text electronic data.

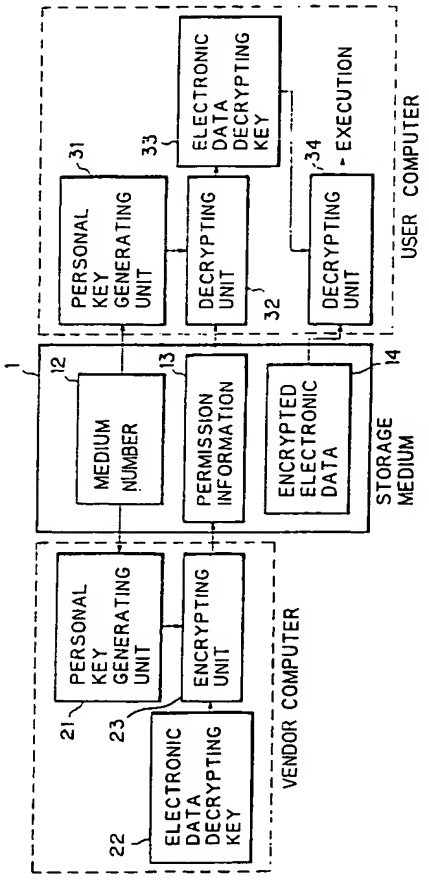


Fig. 2

EP 0 561 685 A2

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to an electronic data protection system, and more particularly, it relates to an electronic data protection system for protecting electronic data, for example, software used for a computer and data published electronically, from being illegally copied by a third party.

2. Description of the Related Art

Recently, various computers, electronic publishing and the like, which utilize electronic data, are widely used in various fields. In general, software and electronic data are protected by copyright in various countries.

However, it is relatively easy for a third party to illegally copy electronic data. As a result of illegal copying, a vendor of electronic data suffers significant damage in that he cannot derive legitimate benefits. As a result of this damage, the cost of electronic data, i.e., the software and electronically published data rises so that users also suffer due to increased prices.

Accordingly, it is necessary to provide a protection system for electronic data, such as software and electronically published data, in addition to protection by means of copyright.

SUMMARY OF THE INVENTION

The object of the present invention is to provide an electronic data protection system enabling certain protection of electronic data, such as software used for a computer and electronically published data from illegal copying by a third party.

In accordance with the present invention, there is provided an electronic data protection system for protecting electronic data from illegal copying by a third party, the system including:

a storage medium for storing an encrypted electronic data, a medium number and an encrypted permission information; a vendor computer having a personal key generating unit for generating a medium key based on the medium number, an electronic data decrypting key, and an encrypting unit for encrypting the electronic data decrypting key based on the medium key to generate the encrypted permission information; and a user computer having a personal key generating unit for generating a medium key based on the medium number, a decrypting unit for decrypting the encrypted permission information based on the medium key to generate an electronic data decrypting key which is the same as the electronic data decrypting key of the vendor computer, and a decrypting unit for decrypting the encrypted electronic data based on the electronic data decrypting key to generate plain text (unencrypted) electronic data.

In a preferred embodiment, the electronic data is software used in a computer.

In another preferred embodiment, the electronic data is electronically published data.

In still another preferred embodiment, the storage medium stores a plurality of encrypted electronic data, and each encrypted electronic data has a different electronic data decrypting key; the vendor computer encrypts only an electronic decrypting key for the encrypted electronic data permitted by a vendor by using the medium key, and stores the encrypted electronic data key on the storage medium as the encrypted permission information; and the user computer decrypts the encrypted electronic data corresponding to the encrypted permission information.

In still another preferred embodiment, the user computer writes the medium number on the storage medium in an un-rewritable form.

In still another preferred embodiment, the vendor computer stores the encrypted permission information on a different storage medium, and the different storage medium is supplied for use with the user computer.

In still another preferred embodiment, the vendor computer transfers the encrypted permission information to the user computer through a transmission line, and the user computer decrypts the encrypted electronic data from the storage medium based on the encrypted permission information.

In still another preferred embodiment, the vendor computer sends the encrypted permission information to the user in a document, and the user computer decrypts the encrypted electronic data from the storage medium based on the encrypted permission information described in the document.

In still another preferred embodiment, the storage medium is an optical magnetic disk, or a partially embossed optical disk.

In still another preferred embodiment, the vendor computer further comprises a software encrypting key management table including software names and encrypting keys corresponding to respective software names.

In still another preferred embodiment, the user computer further comprises a software decrypting key management table including software names and decrypting keys corresponding to respective software names.

BRIEF EXPLANATION OF THE DRAWINGS

In the drawings:

Fig. 1 is a schematic block diagram of a conventional electronic data protection system;

Fig. 2 is a principal view of the present invention;

Fig. 3 shows one embodiment of the present in-

vention;

Fig. 4 is a flowchart of a storage process of software according to the present invention;

Figs. 5A and 5B are explanatory views for one example of the encrypting process;

Fig. 6 shows a software encrypting key management table according to an embodiment of the present invention;

Figs. 7A and 7B are flowcharts for generating permission information;

Fig. 8 is an explanatory view of generation of the permission information;

Figs. 9A and 9B are flowcharts for a decrypting process of software;

Figs. 10A, 10B, and 10C are explanatory views of a program as electronic data;

Fig. 11A, 11B, and 11C are explanatory views of data as electronic data;

Fig. 12 is an explanatory view of a ROM/RAM mixed type optical magnetic disk;

Fig. 13 is an explanatory view for permission information stored in another storage medium; and

Fig. 14 is an explanatory view of multiple software written on one storage medium.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Before describing the preferred embodiments, a conventional art and its problems will be explained below.

Figure 1 is a schematic block diagram of a conventional electronic data protection system, particularly, a software protection system. The feature of the conventional software protection system lies in generation of permission information 72 by using a user's personal number 91 as explained in detail below.

As shown in the drawing, a software storage medium 71, for example, an optical magnetic disk, a CD-ROM, a floppy disk and the like, is provided by a vendor to a user. That is, the vendor provides the software storage medium 71 storing the software to the user. The software storage medium 71 includes the permission information 72 and encrypted software 73. The vendor computer includes a personal key generating unit 81, a software decrypting key 82 and an encrypting circuit 83. Further, the user computer includes the user's personal number 91, a personal key generating unit 92, a decrypting circuit 93, a software decrypting key 94, and a decrypting circuit 95.

The software is stored in the software storage medium 71 after encryption. A user's personal key is generated by using the user's personal number 91, and the software decrypting key 82 is decrypted by the user's personal key so that the encrypting circuit 83 encrypts the software decrypting key 82 to generates the permission information. The permission information is stored on the software storage medium 71.

The user buys the software storage medium 71 including the permission information 72 and the encrypted software 73, and the user computer decrypts the encrypted software 73 so that it is possible to execute the unencrypted program.

In the vendor computer, the personal key generating unit 81 generates the personal key based on the user's personal number 91, for example, an apparatus number of a computer. The software decrypting key 82 decrypts the encrypted software 73. The encrypting circuit 83 generates the permission information 72 for storage on the software storage medium 71.

In the user computer, the personal key generating unit 92 receives the user's personal number 91 and generates the user's personal key. The decrypting circuit 93 decrypts the permission information 72 from the software storage medium 71 based on the personal key 81. The software decrypting key 94 is input to the decrypting circuit 95, the decrypting circuit 95 decrypts the encrypted software 73 to generate the plain text software. As a result, the plain text software is loaded into a main storage of the user computer.

An explained above, the conventional software protection system utilizes a user's personal number (or, an apparatus number of a computer) and a personal key.

In use of the personal number for the computer, the execution for the computer is applied by the permission information 72 so that only that computer can execute the plain text software. Accordingly, the user cannot utilize a different computer even if he is authorized. Further, it is impossible to transfer such plain text software to a third party.

In use of the personal number for portable hardware, it is necessary to provide an interface between the hardware and the computer so that the total cost for protection rises.

As a feature of the present invention, a storage medium for storing electronic data includes a medium number. A vendor authorizes use of the medium number. According to the present invention, it is possible to access only electronic data stored on the medium authorized by the vendor.

Figure 2 is a principal view of the present invention. In Fig. 2, the storage medium 1 includes a medium number 12, permission information 13, and encrypted electronic data 14. In the present invention, the electronic data includes computer software and electronically published data, as explained below. The vendor computer includes a medium personal key generating unit 21, an electronic data decrypting key 22, and a decrypting unit 23. The user computer includes a medium personal key generating unit 31, a decrypting unit 32, an electronic data decrypting key 33, and a decrypting unit 34.

The personal key generating units 21 and 31 generate a medium key based on the medium number 12. The encrypting unit 23 encrypts the electronic data decrypting key 22 based on the medium key.

The decrypting unit 32 decrypts the permission information 13 based on the medium key, and generates the electronic data decrypting key 33. Further, the decrypting unit 34 decrypts the encrypted electronic data 14 based on the electronic data decrypting key 33, and generates the plain text electronic data.

In the present invention, briefly, only the medium number 12 and the encrypted electronic data 14 are previously stored in the storage medium 1.

In the vendor computer, the personal key generating unit 21 generates the medium key based on the medium number, the encrypting unit 23 encrypts the electronic data decrypting key 22 based on the medium key, and the encrypting unit 23 writes the encrypted data onto the storage medium 1 as the permission information 13.

In the user computer, the personal key generating unit 31 generates the medium key based on the medium number 12 of the storage medium 1, the decrypting unit 32 decrypts the permission information 13 based on the personal key, and generates an original electronic data decrypting key 33, and decrypting unit 34 decrypts the encrypted electronic data 14 based on the original electronic data decrypting key 33, and provides the plain text electronic data.

Further, a different electronic data decrypting key 22 is provided for every encrypted electronic data 14. Only the electronic data decrypting key 22 of the encrypted electronic data 14, in which the use is permitted in the vendor computer, is encrypted by the medium key, and stored on the storage medium 1 as the permission information 13. Further, only the encrypted electronic data corresponding to this permission information 13 stored on the storage medium 1 is decrypted in the user computer to provide the plain text electronic data.

Still further, only the medium number 12 may be written onto the storage medium 1 in the un-rewritable form by the user computer. Still further, only the permission information 13 may be stored on another medium, for example, a floppy disk, by the vendor computer, and be provided for use in the user computer. Still further, the vendor may transfer the permission information 13 to the user computer through a transmission line, so the user computer can decrypt the encrypted electronic data 14 based on the permission information 13 to provide the plain text electronic data.

In this case, as the encrypted electronic data 14, there are software for operating various computers, or various types of data (character data, image data, sound data), and these are encrypted to protect content thereof. Accordingly, since the medium 1 storing the encrypted electronic data 14 includes the medium number in the un-rewritable form, the vendor computer allows use of the electronic data having the medium number. Accordingly, it is possible to use only the encrypted electronic data 14 stored on the correct medium 1 and authorized by the vendor. As a result, it is possible to transfer the stored electronic data to another user so that it is possible to use this medium 1 in another computer.

Figures 3 to 14 are explanatory views for embodiments of the present invention. As one example of electronic data shown in Fig. 2, an explanation is given next for software used for a computer.

Figure 3 shows one embodiment of the present invention. In Fig. 3, the software storage medium 11, for example, an optical magnetic disk (particularly, a disk having a capacity in the range of several hundred M-byte to several G-byte), is a medium for storing software authorized by a vendor for use by a user. The software storage medium 11 stores the medium number 12 in an un-rewritable form, the permission information 13 granting permission for use of the software to the user, and the encrypted software 15.

The medium number 12 is a particular number for the medium 11, and this medium number cannot be rewritten (i.e., is un-rewritable) by the user. That is, this medium number 12 is written into an area which the user cannot rewrite. This area can be managed in various forms. For example, this area may be managed by an Operating System (OS). Further, it may be possible to arranged that this area cannot be rewritten by means of the Operating System.

The permission information 13 is provided from the vendor to the user to permit use of the software. In this case, this permission information 13 incorporates encrypted data for decrypting the encrypted software 15 (see, Figs. 7 and 8). The encrypted software 15 is shown in Figs. 4 to 6.

The vendor computer includes the personal key generating unit 21, the software decrypting key 24, and the encrypting unit 23. The personal key generating unit 21 generates the medium key based on the medium number 12 read from the software storage medium 11 (see, Fig. 7). The encrypting unit 23 encrypts the software decrypting key 24 based on the medium key generated by the personal key generating unit 21. The data encrypted by the encrypting unit 23 is stored in the software storage medium 11 as the permission information 13.

The user computer includes the personal key generating unit 31, the decrypting unit 32, the software decrypting key 35, and the decrypting unit 34. The personal key generating unit 31 generates the medium key based on the medium number 12 read from the software storage medium 11 (see, Fig. 7). The decrypting unit 32 decrypts the permission information 13 read from the software storage medium 11 based on the medium key generated by the personal key generating unit 31, and generates the software

decrypting key 35 (see, Fig. 9). The decrypting unit 34 decrypts the encrypted software 15 read from the software storage medium 11 based on the software decrypting key 35, and generates the plain text software (see, Fig. 9). The user computer then executes the plain text software.

Figure 4 is a flowchart of a storage process of the software according to the present invention. This flowchart shows the storage process of the encrypted software 15 and the encrypted permission information 13. In step S1, the vendor generates the software, for example, a job program, etc., and stores the generated software onto the software storage medium. In step S2, the vendor generates the software encrypting key. In step S3, the software encrypting key is stored in an encrypting key management table (see, Fig. 6) corresponding to each software. That is, the encrypting key generated by step S2 is stored in the encrypting key management table in correspondence with the name of the software generated by step S1.

In step S4, the software encrypting key corresponding to the software designated by the vendor is taken from the encrypting key management table. In step S5, the vendor encrypts the plain text software by using the software encrypting key taken from the encrypting key management table to generate the encrypted software. As shown in Fig. 5, a main body of the software is encrypted by the encrypting key so as to generate the encrypted software body by using, for example, a data encryption standard (DES) which is widely used in the United States. As shown in Fig. 5B, in the DES, the main body of the software having 64-bit train is encrypted to the same 64-bit train, but having a different bit order.

In step S6, the encrypted software is stored in the storage medium of the vendor (or, a manufacturer of the medium) so that it is possible to hold the once encrypted software. Accordingly, in a subsequent use of the software, the encrypting process can be omitted by the vendor since the software held on the medium is used again. In step S7, the encrypted software is stored on the software storage medium 11. In step S8, the vendor determines whether or not the encrypting process for the software is completed and the encrypted software is stored on the storage medium. When the result is "YES", the encrypting process is completed by the vendor.

When the result is "NO", the encrypting process returns to the step S7 and sequentially stores the encrypted software having the name of the software designated. As explained above, the encrypted software is stored on the software storage medium 11.

Figures 5A and 5B are explanatory views of one example of the encrypting process. A header portion H includes a software name as an identifier, and a main body of the software S includes the plain text software. The header portion H is not encrypted, and the main body of the software S is encrypted by the encrypting key K. The encryption is performed, for example, using the DES (Data Encryption Standard) as shown in Fig. 5B. The DES is already known in the United States.

As shown in Fig. 5B, according to the encrypting process of the DES, the 64-bit bit train of the plain text software is converted to the same 64-bit train as above, but having a different bit order. The decrypting unit decrypts the encrypted 64-bit train to the plain text software by using the DES.

Figure 6 shows a software encrypting key management table according to an embodiment of the present invention. As shown in the drawing, the software encrypting key management table 4 is formed by the name of the software and the encrypting key corresponding to the name of the software. An escape character "ENC" is attached to each software name to indicate that the corresponding software is already encrypted. Further, the encrypted key is formed by a 64-bit bit. train as explained above.

(1) Regarding the plain text software to be stored onto the storage medium, the software encrypting key is taken from the software encrypting key management table.

(2) The encrypting circuit 41 encrypts the plain text software based on the software encrypting key obtained by the above (1).

(3) The encrypted software is stored on the software storage medium 11 as the encrypted software 15.

The above steps are repeated for all plain text software designated by the vendor. As explained above, since once encrypted software is held by the vendor, this encrypted software is stored again on the software storage medium when another user requests this plain text software. The medium number 12 is provided for only the corresponding storage medium, and is written onto the medium 11 in an un-rewritable form (i.e., the personal number cannot be rewritten).

Further, the encrypting key stored in the software encrypting key management table 4 coincides with the decrypting key when using an object key number as an encrypting algorithm. As explained above, the software encrypting key corresponding to the plain text software is taken from the software encrypting key management table, the plain text software is encrypted by the software encrypting key to generate the encrypted software, and the encrypted software is stored in the software storage medium 11.

Figures 7A and 7B are flowcharts for generating permission information, and Figure 8 is an explanatory view of generation of the permission information. The flowchart explains that the permission information 13 is generated for the software to be authorized and stored on the software storage medium.

In step S11, the name of the software to be authorized is input to the software decrypting key management table 5. In step S12, the software decrypting key is loaded from the software decrypting key management table 5. As shown in Fig. 8, the decrypting key corresponding to the name of the software to be authorized is sent from the software decrypting key management table 5 to the encrypting circuit 33.

In step S13, the medium number 12 is loaded from the software storage medium 11. In step S14, the medium key is generated in the personal key generating circuit 211. That is, as shown in Fig. 7B, the medium number (plain text, i.e., unencrypted) is encrypted by a secret key (or, a secret algorithm) so that the medium key (encrypted) is generated. In general, in use of the DES, a secret key is used, and in use of a secret algorithm, the secret key is not used.

In step S15, the software decrypting key (plain text, i.e., unencrypted) is encrypted by the medium key so that the permission information (encrypted) is generated. The DES is used for the above encrypting process. In step S16, the permission information encrypted by the step S15 is stored in the software storage medium 11.

As explained above, briefly, the encrypted software is stored on the software storage medium 11, the medium key 12 is read therefrom, the software decrypting key is encrypted by the medium key to generate the encrypted permission information 13, and the encrypted permission information 13 is stored on the software storage medium 11. Accordingly, the encrypted software 15 and the encrypted permission information 13 are stored on the software storage medium 11.

In Fig. 8, the software decrypting key management table 5 is provided for managing the software decrypting key which is used when the encrypted software 15 is decrypted to derive the plain text software. The software decrypting key is managed in correspondence with the software name. The software decrypting key management table 5 stores the decrypting key having the same structure as the software encrypting key management table 4.

(1) When the vendor sells permission information to a user, the medium number 12 is read from the software storage medium 11. The personal key generating circuit 211 receives this medium number and generates the medium key (see, step S14 of Fig. 7A).

(2) Next, the software decrypting key corresponding to the software to be sold is taken from the software decrypting key management table 5, and this key is input to the encrypting circuit 231. This software decrypting key is encrypted by the personal key in the encrypting circuit 231 so that the encrypting circuit 231 generates the permission information 13. The permission information 13 includes the software name having the escape character ENC and the encrypted permission information, and this permission information 13 is stored on the software storage medium 11. In this case, the software decrypting key and the algorithm (or, the secret key) are protected by a known safety means (not shown).

As explained above, the vendor generates the medium key based on the medium number 12 read from the software storage medium 11, encrypts the software decrypting key based on the medium key, and stores this software decrypted key into the software storage medium 11 as the permission information 13.

Figures 9A and 9B are a flowcharts of a decrypting process of the software. The user buys the software storage medium 11 and mounts it in the user computer. The software of the storage medium 11 is loaded into a main memory to execute the program.

In step S21, the user computer receives an instruction to execute the software. In step S22, the medium number 12 is taken from the software storage medium 11. In step S23, the medium key (unencrypted) is encrypted by the secret key (or, the algorithm) to generate the encrypted medium key. In step S24, the encrypted permission information is decrypted by th medium key so that the plain text software decrypting key is generated.

In step S25, the encrypted software 15 is read from the software storage medium 11. In step S26, the encrypted software is decrypted by the software decrypting key so that the plain text software is generated. In step S27, the user computer executes the plain text software.

As explained above, briefly, the medium key is generated based on the medium key 12 from the software storage medium 11, the permission information 13 is decrypted based on the medium key to generate the software decrypting key 35, and the encrypted software is decrypted by the software decrypting key 35 to generate the plain text software.

Figures 10A, 10B, and 10C are explanatory views of a program as electronic data. Figure 10A shows an entire structure, Figure 10B shows a flowchart, and Figure 10C is an explanatory view of execution of the software. In Fig. 10A, an optical magnetic disk 6 corresponds to the software storage medium 11 of Figs. 2 and 3, and stores the medium number 12, the permission information 13 and the encrypted program 16. The user buys the optical magnetic disk 6 and mounts this disk in an optical magnetic apparatus. As other examples, an optical disk, a CD-ROM, a floppy disk, a hard disk, a magnetic tape, a cassette tape, and the like are known as storage media.

A program loader 61 has a function of a key generating process (personal key generating process 31) and a decrypting process (decrypting process 32 and 34), and loads the corresponding decrypted program from the optical magnetic disk 6 into a main storage

63 in an execution stage of the program instruction so that the execution state of the program is established. The main storage 63 is formed by a RAM to expand the plain text program which is taken from the optical magnetic disk 6.

In Fig. 10B, in step S31, the program loader 61 receives the execution instruction for the program. In step S32, the program loader 61 loads a program to be executed and decrypts it. In step S33, the plain text program is expanded in the main memory to obtain an executable plain text program. In step S34, the plain text program in the main memory is executed.

In Fig. 10C, a relationship between the software storage medium and the user computer is explained in detail.

(1) The user computer takes the medium number 12 from the software storage medium 11, and sends it to the personal key generating circuit 311 to generate the encrypted medium key (see, step S23 of Fig. 9).

(2) The decrypting circuit 321 receives the permission information 13 from the software storage medium 13, and encrypts it based on the medium key from the generating circuit 311. As a result, the software decrypting key 351 (corresponding to the software decrypting key 35) is obtained by the decrypting circuit 321.

(3) The decrypting circuit 341 receives the encrypted software 15 from the software storage medium 11, and decrypts it based on the software decrypting key 351 to generate the plain text program, then the plain text program is stored in the main storage 63.

As explained above, it is impossible to decrypt the encrypted software 15 in which the permission information 13 is not stored so that it is impossible to execute such a program. Further, if the software storage medium 11 is illegally copied by a third party, since the medium number 12 is not provided or is different, it is impossible to decrypt the correct software decrypting key 351 from the permission information 13. As a result, it is impossible to decrypt the encrypted program so that it is impossible to execute the program. As explained above, in the user computer, an algorithm and a secret key in the personal key generating circuit 311, a software decoding key, and the plain text software are protected by a known safety means.

Figure 11A, 11B, and 11C are explanatory views for electronic data, for example, character data (text), symbols, image data, and sound data. Figure 11A shows an entire structure, Figure 11B shows a flowchart, and Figure 11C is an explanatory view showing execution of the software.

In Fig. 11A, an optical magnetic disk 6 corresponds to the software storage medium 11 of Figs. 2 and 3, and stores the medium number 12, the permission information 13 and the encrypted data 17. The user buys the optical magnetic disk 6 and mounts this disk in an optical magnetic apparatus. As other examples, an optical disk, a CD-ROM, a floppy disk, a hard disk, a magnetic tape, a cassette tape, and the like are known as storage media.

A read/write (R/W) module 64 has a function of a key generating process (personal key generating process 31) and a decrypting process (decrypting process 32 and 34), and stores the corresponding decrypted data from the optical magnetic disk 6 into a main storage 63 in an execution stage of the read instruction. The main storage 63 is formed by a RAM to expand the unencrypted data which is taken from the optical magnetic disk 6.

In Fig. 11B, in step S41, the R/W module 64 executes an application program. In step S42, the R/W module 64 reads the data from the optical magnetic disk. In step S43, the R/W module takes the data and encrypts it. In step S44, the unencrypted data is stored in the main memory 63. In step 45, the data is displayed and reproduced.

In Fig. 11C, a relationship between the data storage medium and the user computer is explained in detail.

(1) The user computer takes the medium number 12 from the data storage medium 111, and sends it to the personal key generating circuit 311 to generate the encrypted medium key (see, step S23 of Fig. 9).

(2) The decrypting circuit 321 receives the permission information 13 from the data storage medium 111, and encrypts it based on the medium key from the generating circuit 311. As a result, the data decrypting key 352 (corresponding to the software decrypting key 35) is obtained by the decrypting circuit 321.

(3) The decrypting circuit 341 receives the encrypted data 15 from the data storage medium 111, and decrypts it based on the data decrypting key 351 to generate the unencrypted data, then the unencrypted data is stored in the main storage 63.

Figure 12 is an explanatory view of a ROM/RAM mixed type optical magnetic disk. The ROM/RAM mixed type optical magnetic disk has un-rewritable area for storing the medium number 12, and has a readable/writable area for the permission information 13, and a read only area/write only area for the encrypted software 15. In the present invention, the medium number is given to the optical magnetic disk to write the medium number into the un-rewritable area.

Figure 13 is an explanatory view of the permission information stored in another storage medium. In this case, the software storage medium previously stores the medium number and the encrypted software. The permission information is stored into another storage medium. This means that the medium number and the encrypted software (or, encrypted

data) are previously written onto, for example, the CD-ROM which has no write area, and the permission information is previously written onto a floppy disk.

Figure 14 is an explanatory view of multiple of software written onto one storage medium, for example, an optical disk and a CD-ROM. At the vendor, the encrypting circuit encrypts a plurality of software decrypting keys 1 to N based on the personal key to generate a plurality of permission information 1 to N each corresponding to each software decrypting key 1 to N. A plurality of permission information 1 to N is stored on the software storage medium.

When the user requests a desired software name from the encrypted softwares 1 to N and informs it to the vendor, the vendor encrypts the software decrypting key, which corresponds to the desired software, by the medium key generated by the medium number, and stores encrypted software onto the software storage medium 11. The user mounts this software storage medium into the computer, and the user computer decrypts the encrypted software to generate the plain text software.

Even if a user tries to use software for which he has no permission number, he cannot use such software because the encrypted software cannot be decrypted. Further, even if the user copies the permission information from another software storage medium 11, the medium number in the software storage medium cannot be copies so that it is impossible to decrypt the software correctly. Accordingly, it is possible to individually sell the software.

As explained above, according to the present invention, the medium number 12 is stored on the storage medium 1 which stores the encrypted electronic data in the un-rewritable form, and permission to use the electronic data is given by the medium key 12. Accordingly, it is possible to use only the encrypted electronic data stored in correct medium 1 as authorized by the vendor so that it is possible to prevent illegal copying by a third party. Further, it is possible to transfer the electronic data stored in the storage medium 1. Still further, it is possible to use the same storage medium in another computer. Still further, it is possible to store a plurality of electronic data in one medium. Still further, it is possible to individually sell the storage medium.

Claims

1. An electronic data protection system for protecting electronic data from illegal copying by a third party, comprising:

a storage medium (1) for storing encrypted electronic data (14), a medium number (12) and encrypted permission information (13);

a vendor computer having a personal key generating means (21) for generating a medium key based on the medium number (12), an electronic data decrypting key (22), and an encrypting means (23) for encrypting the electronic data decrypting key (22) based on the medium key to generate the encrypted permission information (13); and

a user computer having a personal key generating means (31) for generating a medium key based on the medium number (12), a decrypting means (32) for decrypting the encrypted permission information (13) based on the medium key to generate the electronic data decrypting key (33) which is the same as the electronic data decrypting key (22) of the vendor computer, and a decrypting means (34) for decrypting the encrypted electronic data (14) based on the electronic data decrypting key (33) to generate a plain text electronic data.

2. An electronic data protection system as claimed in claim 1, wherein the electronic data is software used for a computer.

3. An electronic data protection system as claimed in claim 1, wherein the electronic data is electronically published data.

4. An electronic data protection system as claimed in claim 1, wherein the storage medium (1) stores a plurality of encrypted electronic data, and each encrypted electronic data has a different electronic data decrypting key; the vendor computer encrypts only electronic decrypting key of the encrypted electronic data authorized by a vendor by using the medium key, and stores the encrypted electronic data key onto the storage medium (1) as the encrypted permission information; and the user computer decrypts the encrypted electronic data corresponding to the encrypted permission information.

5. An electronic data protection system as claimed in claim 1, wherein the user computer writes the medium number (12) onto the storage medium in an un-rewritable form.

6. An electronic data protection system as claimed in claim 4, wherein the user computer writes the medium number (12) onto the storage medium in an un-rewritable form.

7. An electronic data protection system as claimed in claim 1, wherein the vendor computer stores the encrypted permission information onto another different storage medium, and the different storage medium is supplied for use in the user computer.

8. An electronic data protection system as claimed in claim 1, wherein the vendor computer transfers the encrypted permission information to the user computer through a transmission line, and the user computer decrypts the encrypted electronic data from the storage medium based on the encrypted permission information.

9. An electronic data protection system as claimed in claim 1, wherein the vendor computer sends the encrypted permission information to the user in a document, and the user computer decrypts the encrypted electronic data from the storage medium based on the encrypted permission information described in the document.

10. An electronic data protection system as claimed in claim 1, wherein the storage medium is an optical magnetic disk, or a partially embossed optical disk.

11. An electronic data protection system as claimed in claim 1, wherein the vendor computer further comprises a software encrypting key management table including software names and encrypting keys each corresponding to each of the software names.

12. An electronic data protection system as claimed in claim 1, wherein the user computer further comprises a software decrypting key management table including software names and decrypting keys each corresponding to each of the software names.

Fig. 1

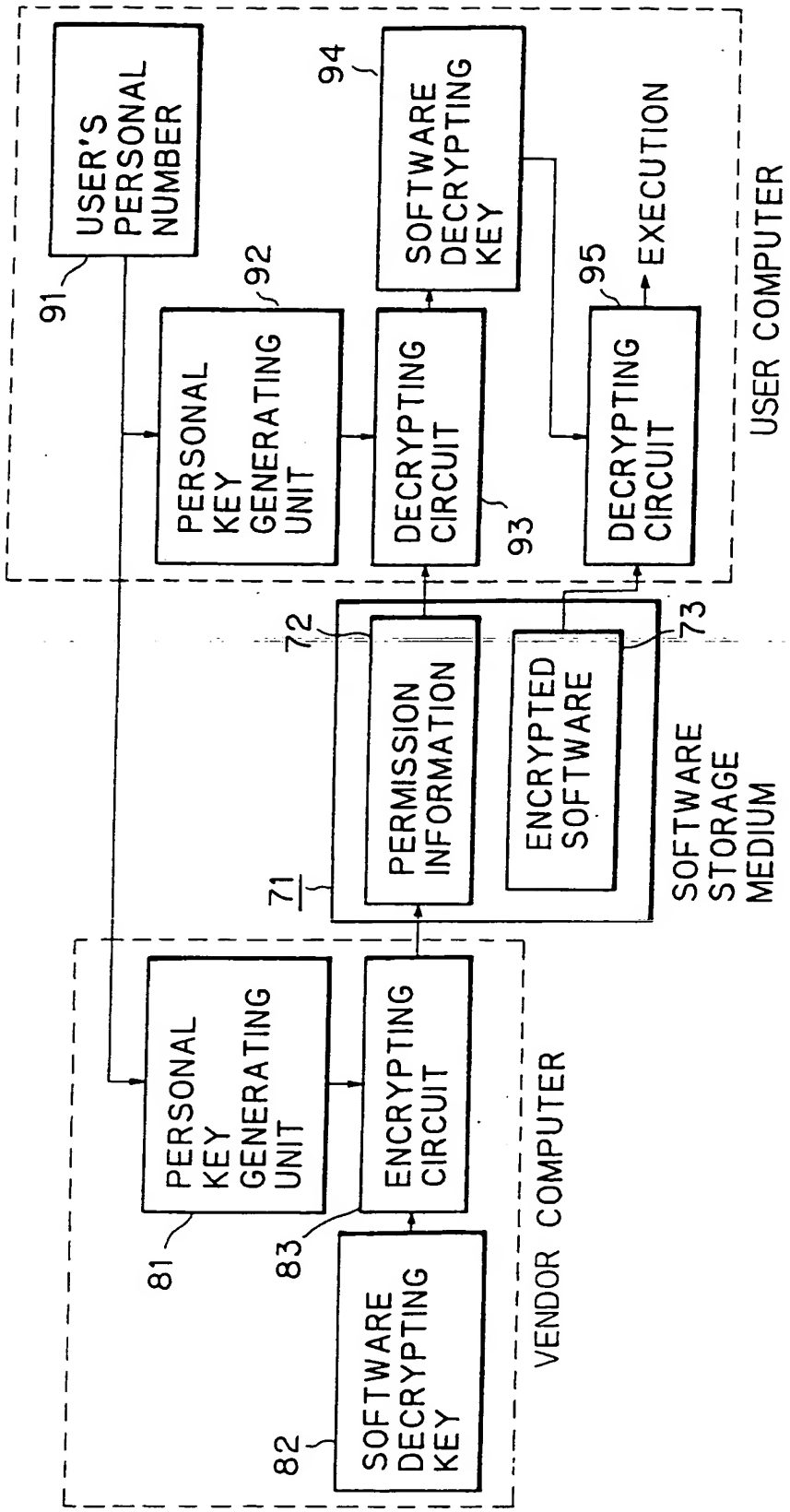

USER'S PERSONAL NUMBER
91
PERSONAL KEY GENERATING UNIT
92
DECRYPTING CIRCUIT
93
SOFTWARE DECRYPTING KEY
94
DECRYPTING CIRCUIT
95
EXECUTION
USER COMPUTER
PERMISSION INFORMATION
72
ENCRYPTED SOFTWARE
73
71
SOFTWARE STORAGE MEDIUM
PERSONAL KEY GENERATING UNIT
81
ENCRYPTING CIRCUIT
83
SOFTWARE DECRYPTING KEY
82
VENDOR COMPUTER

Fig. 2

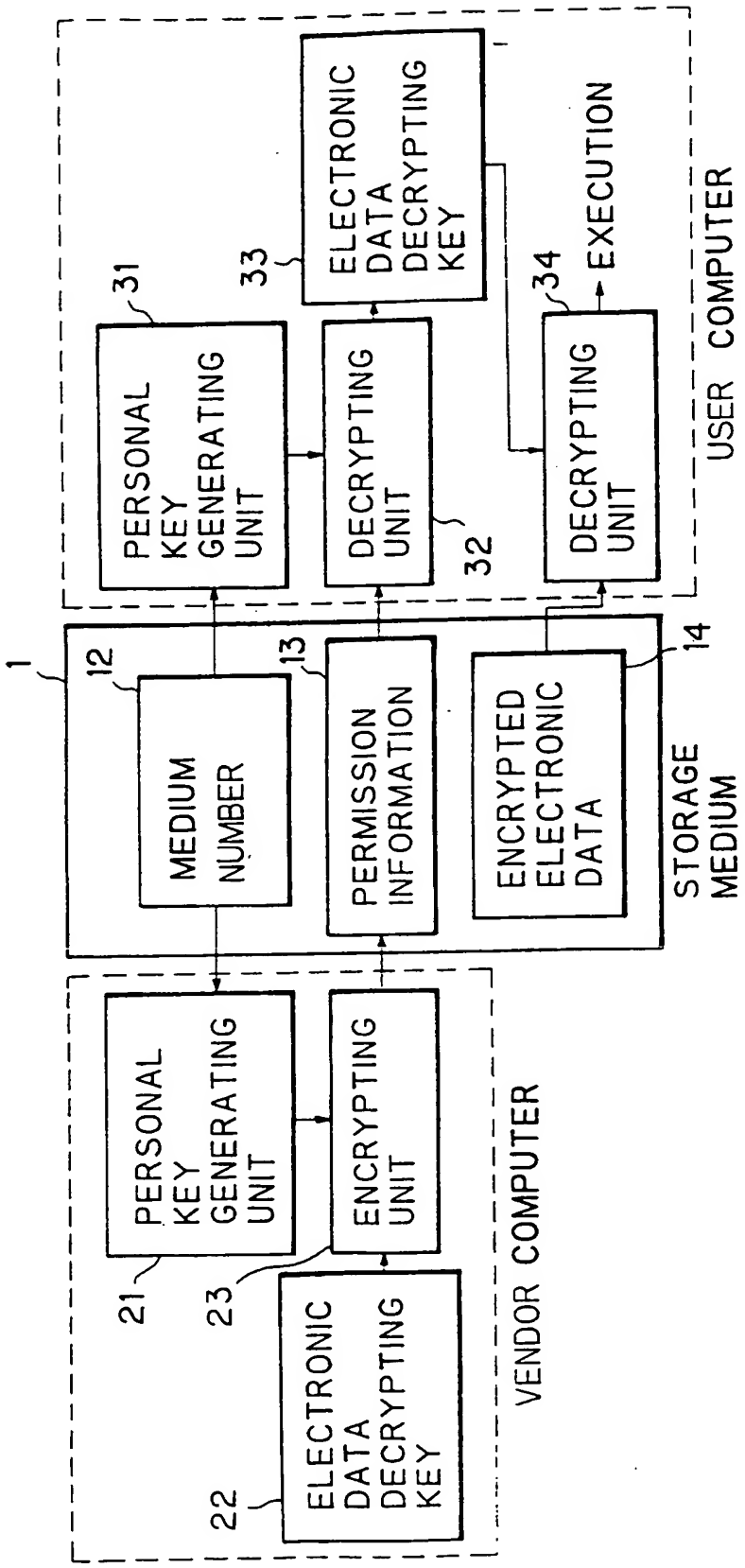

USER COMPUTER
ELECTRONIC DATA DECRYPTING KEY
33
PERSONAL KEY GENERATING UNIT
31
DECRYPTING UNIT
32
DECRYPTING UNIT
34
EXECUTION
1
STORAGE MEDIUM
12
MEDIUM NUMBER
13
PERMISSION INFORMATION
14
ENCRYPTED ELECTRONIC DATA
VENDOR COMPUTER
PERSONAL KEY GENERATING UNIT
21
ENCRYPTING UNIT
23
ELECTRONIC DATA DECRYPTING KEY
22

Fig. 3

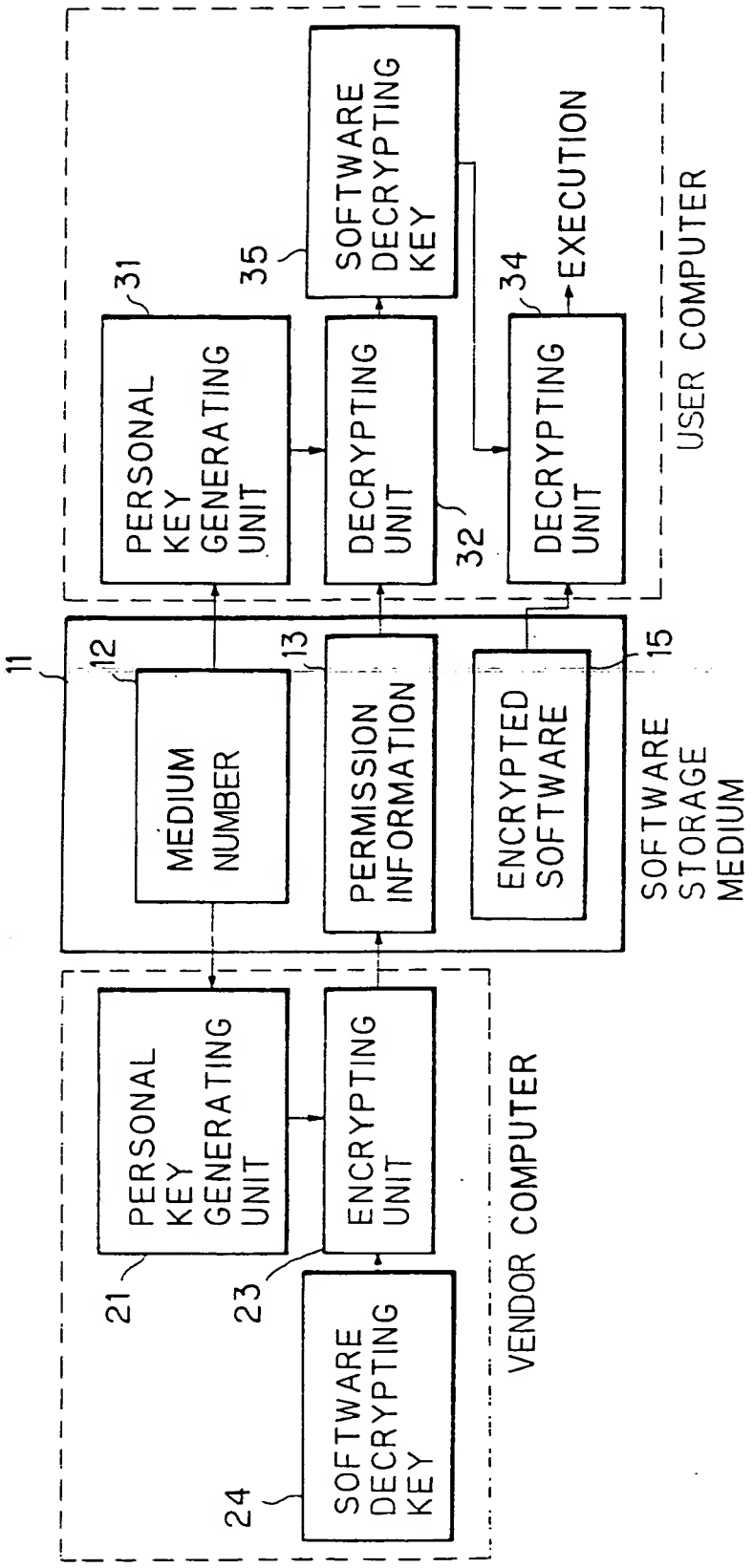

USER COMPUTER
PERSONAL KEY GENERATING UNIT
31
DECRYPTING UNIT
32
SOFTWARE DECRYPTING KEY
35
DECRYPTING UNIT
34
EXECUTION
11
SOFTWARE STORAGE MEDIUM
MEDIUM NUMBER
12
PERMISSION INFORMATION
13
ENCRYPTED SOFTWARE
15
VENDOR COMPUTER
PERSONAL KEY GENERATING UNIT
21
ENCRYPTING UNIT
23
SOFTWARE DECRYPTING KEY
24

Fig. 4

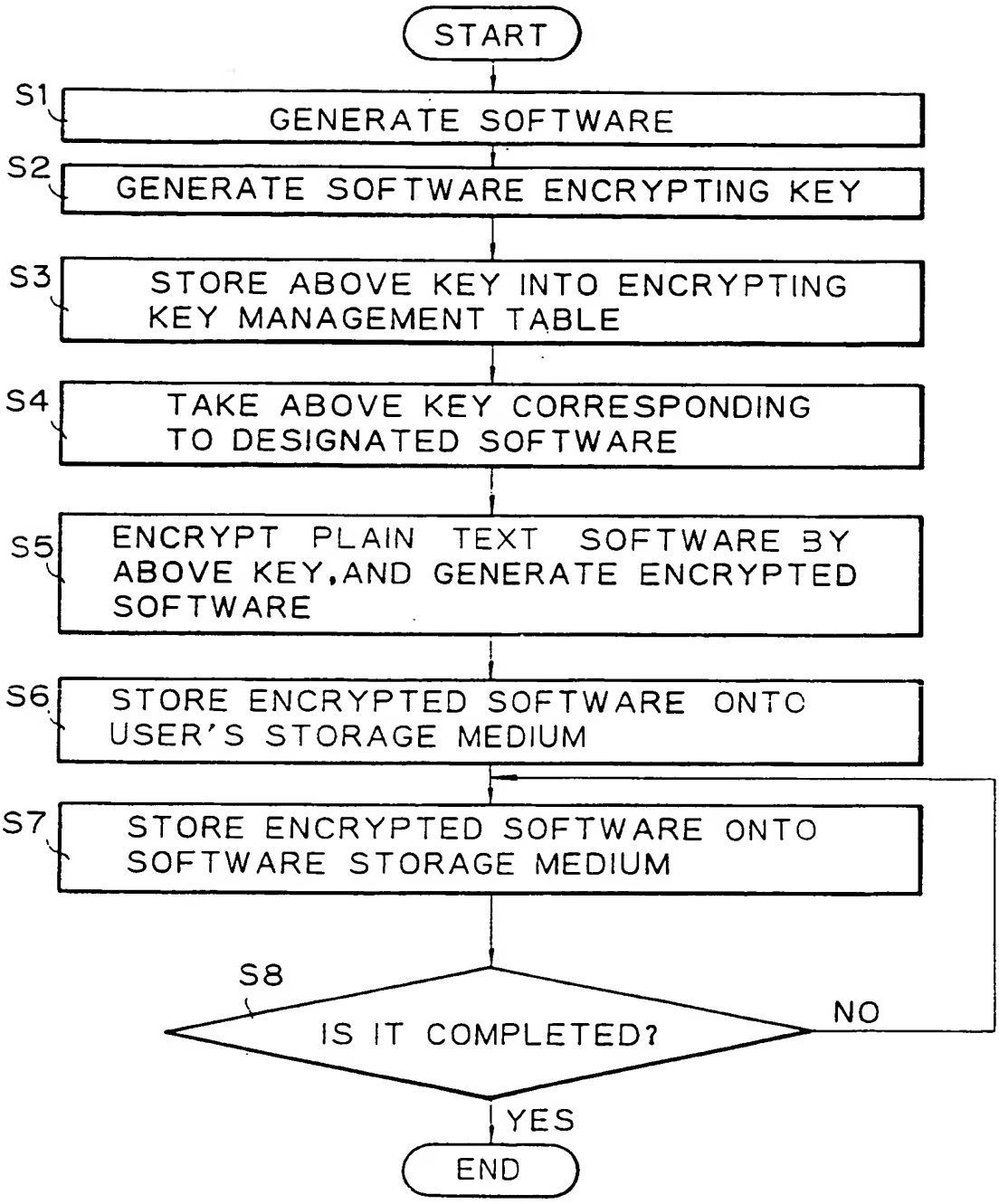

START
S1
GENERATE SOFTWARE
S2
GENERATE SOFTWARE ENCRYPTING KEY
S3
STORE ABOVE KEY INTO ENCRYPTING KEY MANAGEMENT TABLE
S4
TAKE ABOVE KEY CORRESPONDING TO DESIGNATED SOFTWARE
S5
ENCRYPT PLAIN TEXT SOFTWARE BY ABOVE KEY,AND GENERATE ENCRYPTED SOFTWARE
S6
STORE ENCRYPTED SOFTWARE ONTO USER'S STORAGE MEDIUM
S7
STORE ENCRYPTED SOFTWARE ONTO SOFTWARE STORAGE MEDIUM
S8
IS IT COMPLETED?
NO
YES
END

Fig. 5A

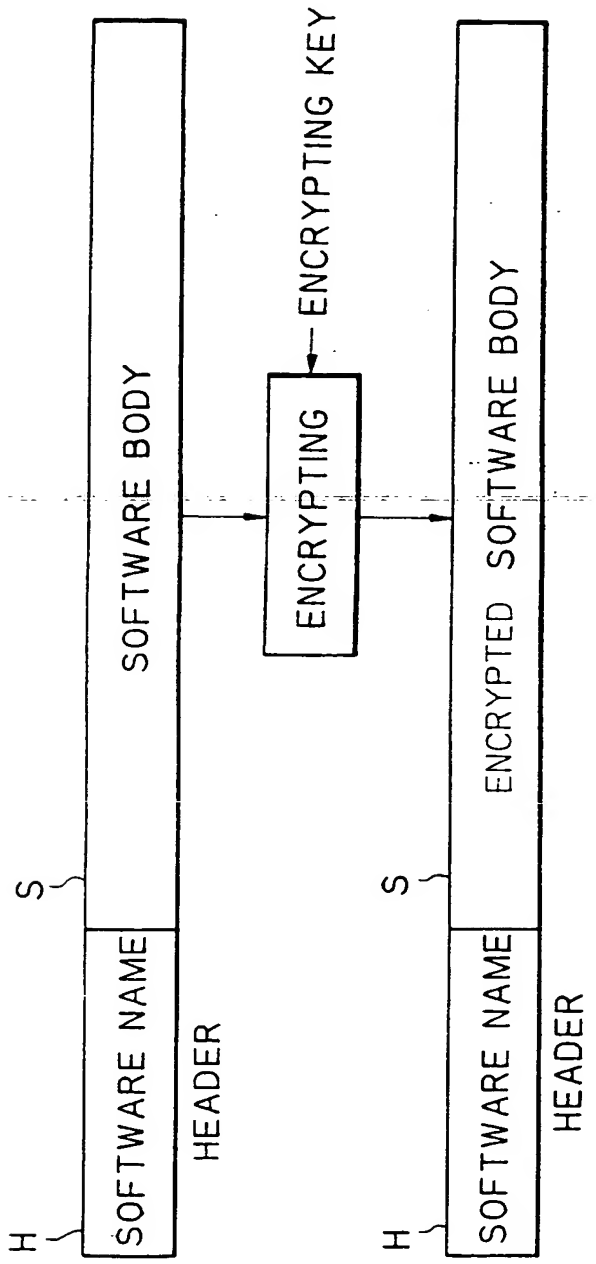

SOFTWARE BODY
ENCRYPTING KEY
ENCRYPTING
ENCRYPTED SOFTWARE BODY
S
S
SOFTWARE NAME
HEADER
H
SOFTWARE NAME
HEADER
H

Fig. 5B

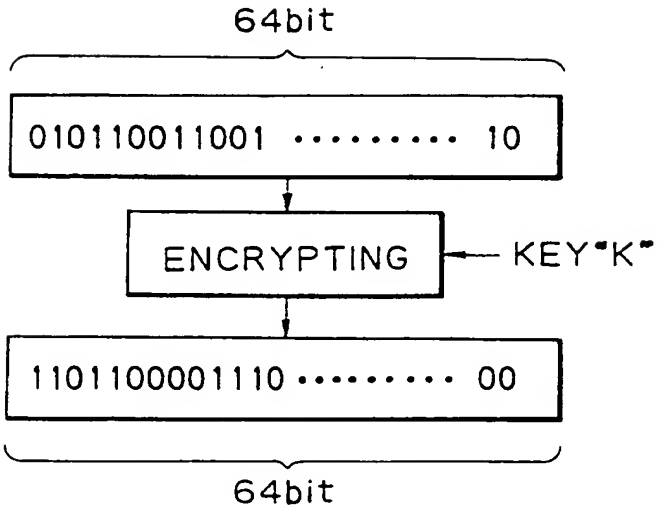

64bit
0101100011001 10
ENCRYPTING
KEY"K"
1101100001110 00
64bit

Fig. 6

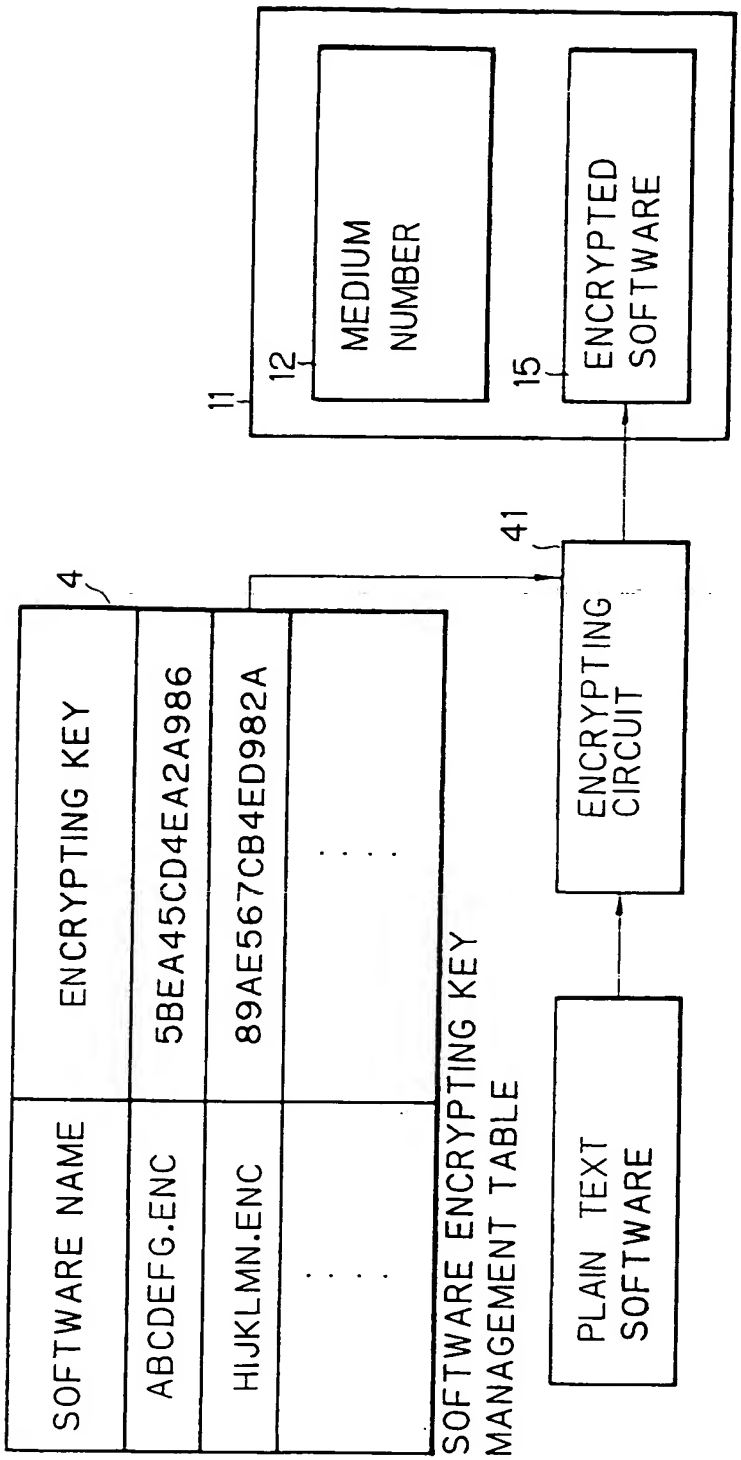

11
12
MEDIUM NUMBER
15
ENCRYPTED SOFTWARE
41
ENCRYPTING CIRCUIT
PLAIN TEXT SOFTWARE
4
SOFTWARE NAME
ENCRYPTING KEY
ABCDEFG.ENC
5BEA45CD4EA2A986
HIJKLMN.ENC
89AE567CB4ED982A
SOFTWARE ENCRYPTING KEY MANAGEMENT TABLE

Fig.7

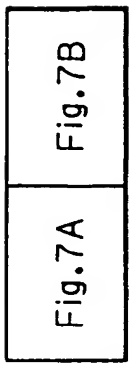


Fig.7A
Fig.7B

Fig. 7A

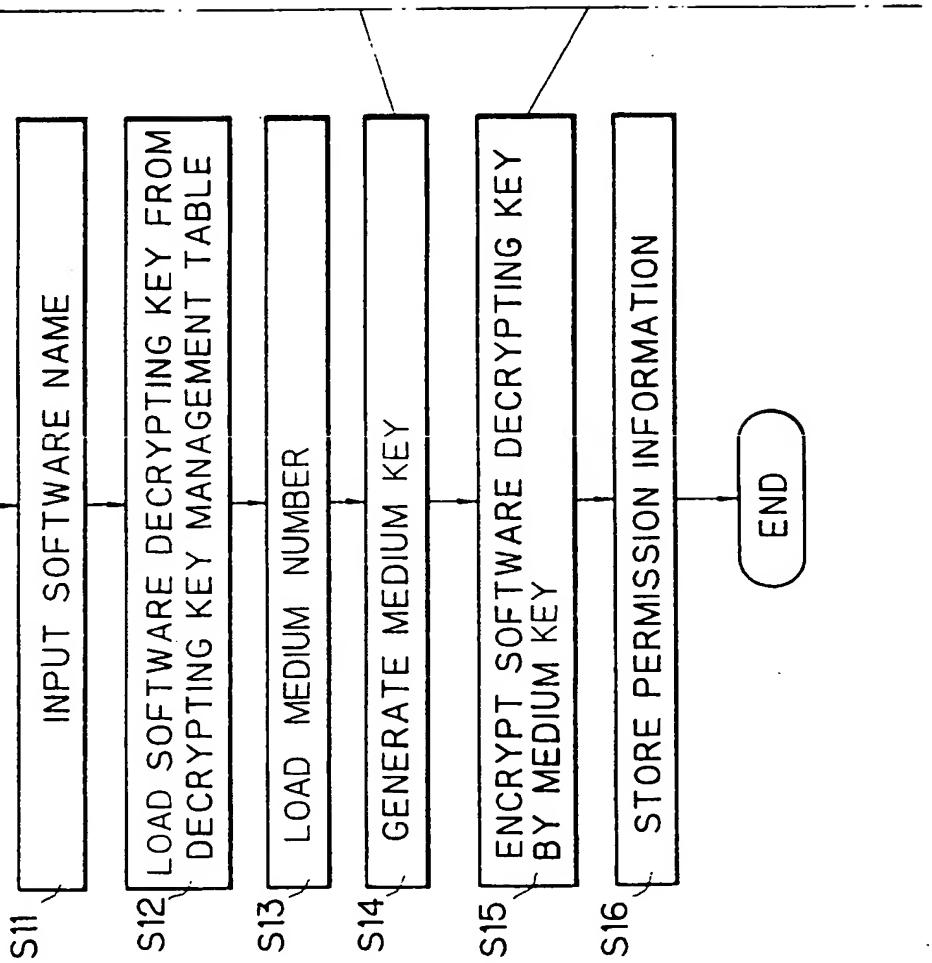

START
INPUT SOFTWARE NAME
LOAD SOFTWARE DECRYPTING KEY FROM DECRYPTING KEY MANAGEMENT TABLE
LOAD MEDIUM NUMBER
GENERATE MEDIUM KEY
ENCRYPT SOFTWARE DECRYPTING KEY BY MEDIUM KEY
STORE PERMISSION INFORMATION
END
S11
S12
S13
S14
S15
S16

Fig. 7B

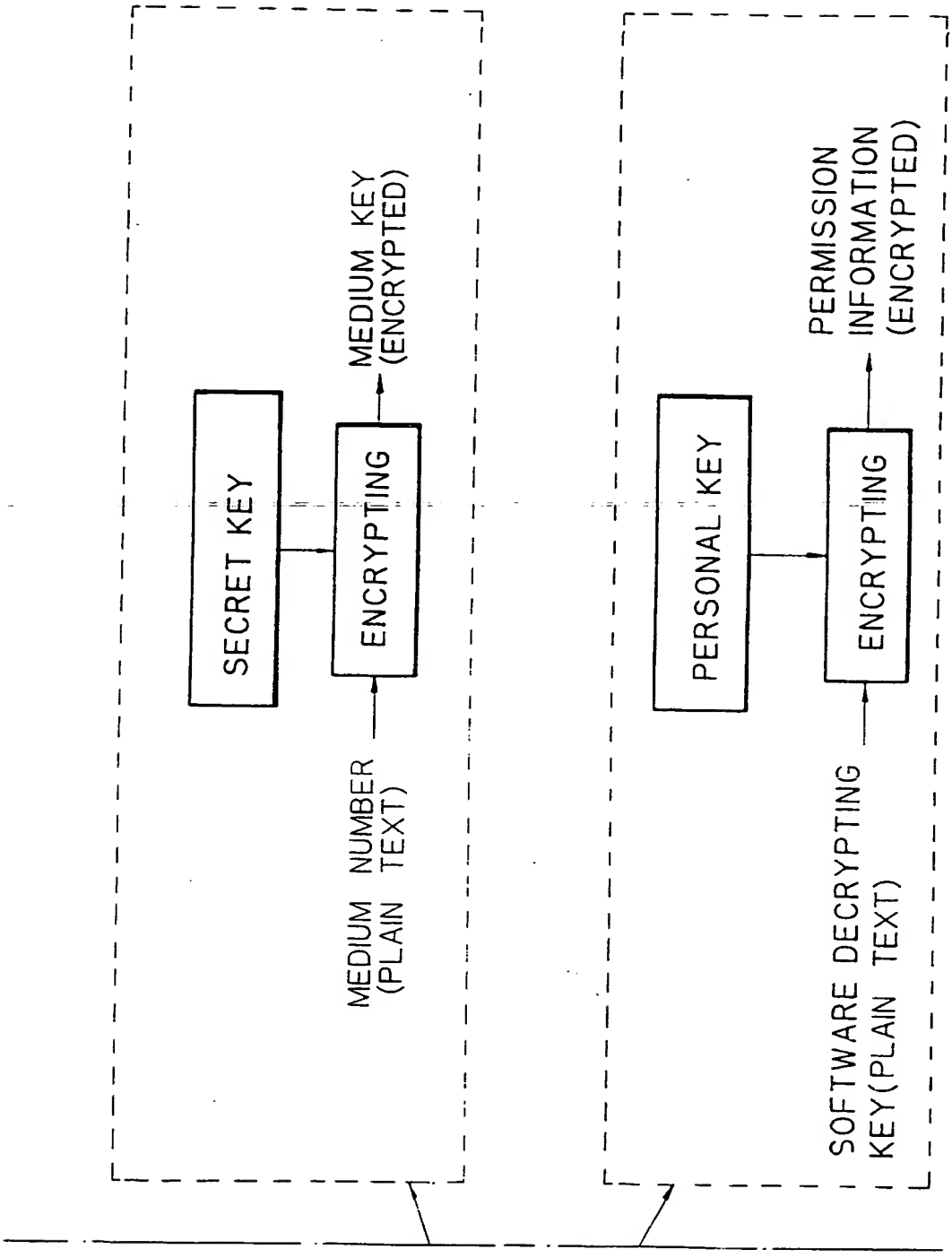

SECRET KEY
ENCRYPTING
MEDIUM KEY (ENCRYPTED)
MEDIUM NUMBER (PLAIN TEXT)
PERSONAL KEY
ENCRYPTING
PERMISSION INFORMATION (ENCRYPTED)
SOFTWARE DECRYPTING KEY(PLAIN TEXT)

Fig. 8

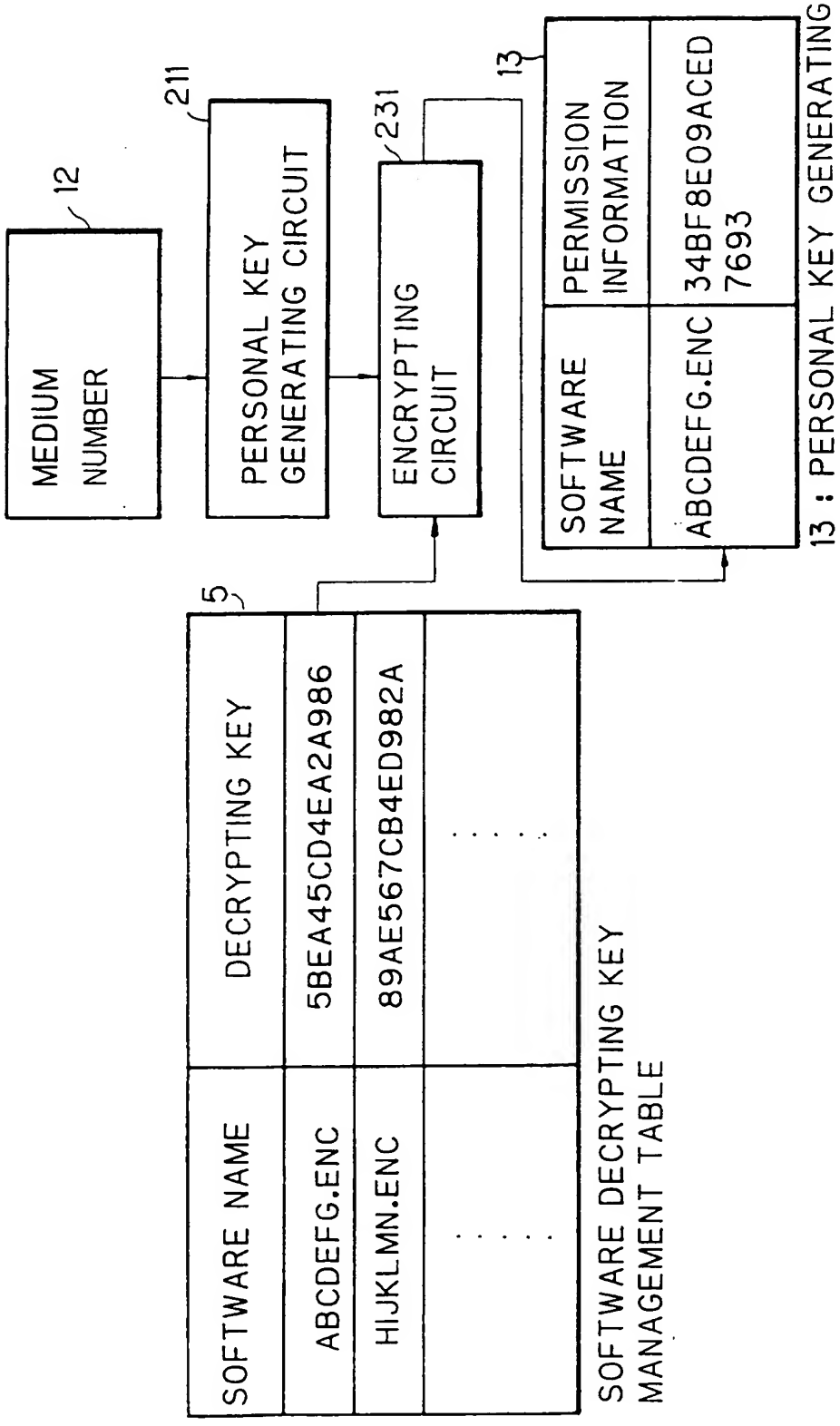

MEDIUM NUMBER
12
211
PERSONAL KEY GENERATING CIRCUIT
231
ENCRYPTING CIRCUIT
13
SOFTWARE NAME
PERMISSION INFORMATION
ABCDEFG.ENC
34BF8E09ACED 7693
13 : PERSONAL KEY GENERATING CIRCUIT
5
SOFTWARE NAME
DECRYPTING KEY
ABCDEFG.ENC
5BEA45CD4EA2A986
HIJKLMN.ENC
89AE567CB4ED982A
SOFTWARE DECRYPTING KEY MANAGEMENT TABLE

Fig.9

Fig.9A	Fig.9B

Fig. 9A

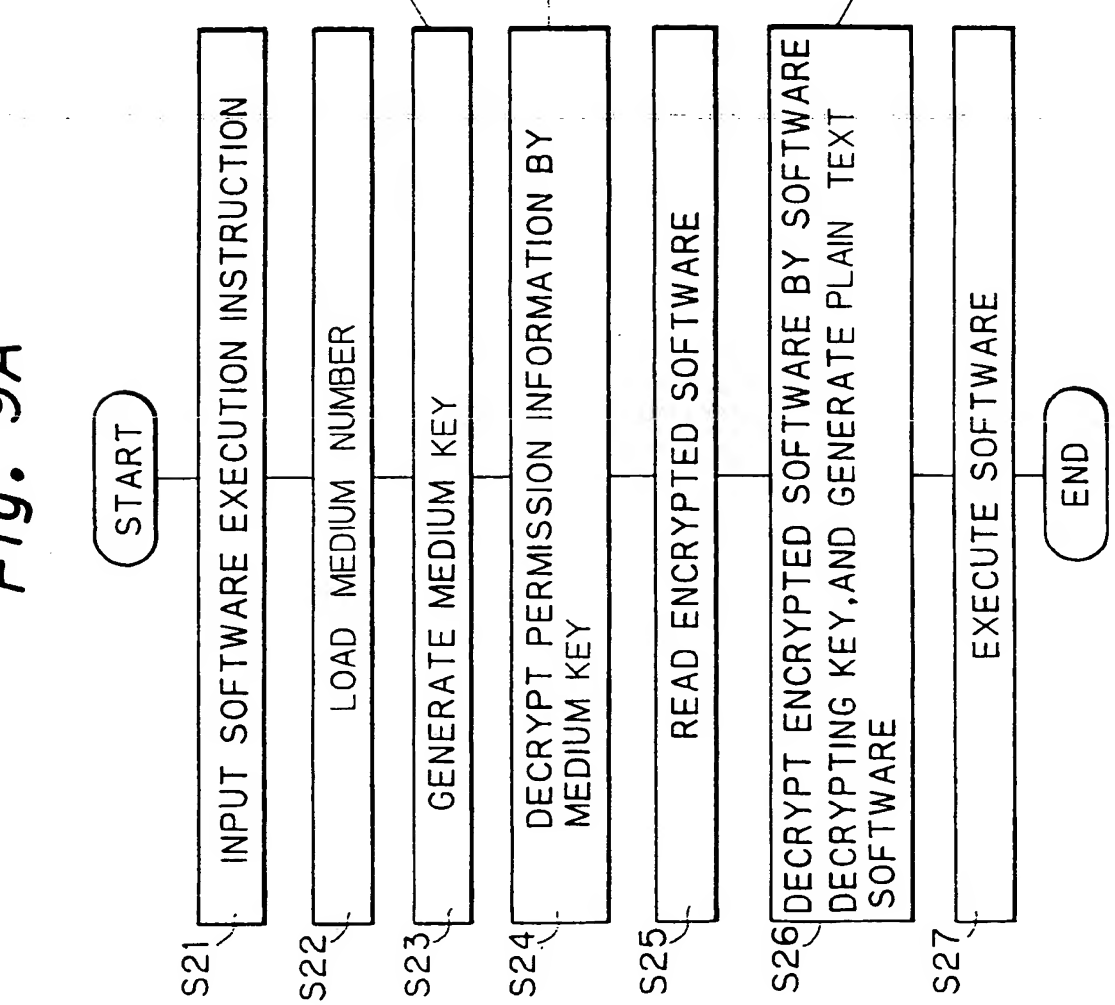


Fig. 9B

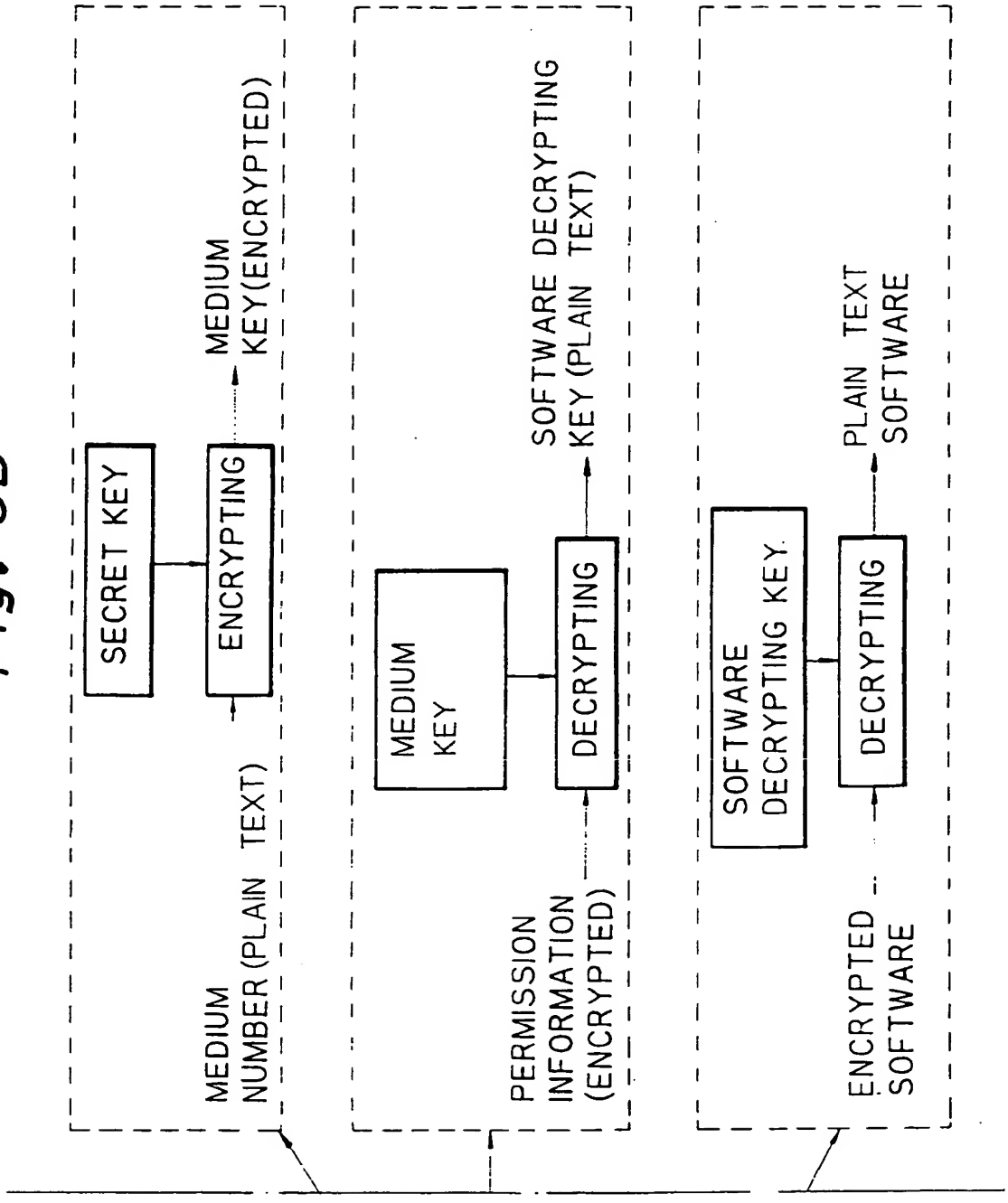

SECRET KEY
ENCRYPTING
MEDIUM KEY(ENCRYPTED)
MEDIUM NUMBER (PLAIN TEXT)
MEDIUM KEY
DECRYPTING
SOFTWARE DECRYPTING KEY (PLAIN TEXT)
PERMISSION INFORMATION (ENCRYPTED)
SOFTWARE DECRYPTING KEY
DECRYPTING
PLAIN TEXT SOFTWARE
ENCRYPTED SOFTWARE

Fig. 10A

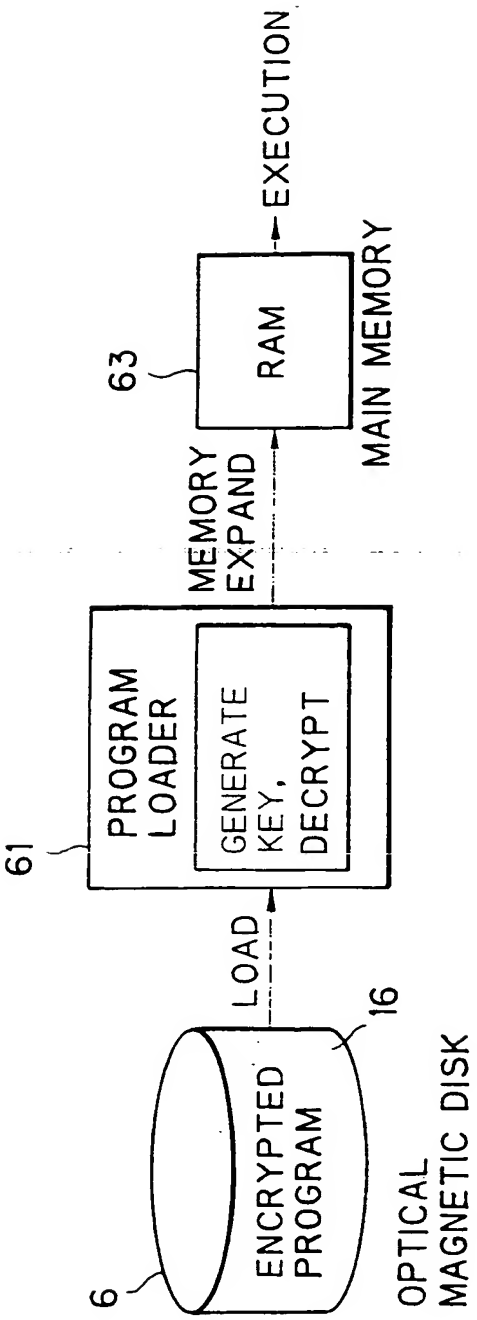

6
ENCRYPTED PROGRAM
16
OPTICAL MAGNETIC DISK
LOAD
61
PROGRAM LOADER
GENERATE KEY, DECRYPT
MEMORY EXPAND
63
RAM
MAIN MEMORY
EXECUTION

Fig. 10B

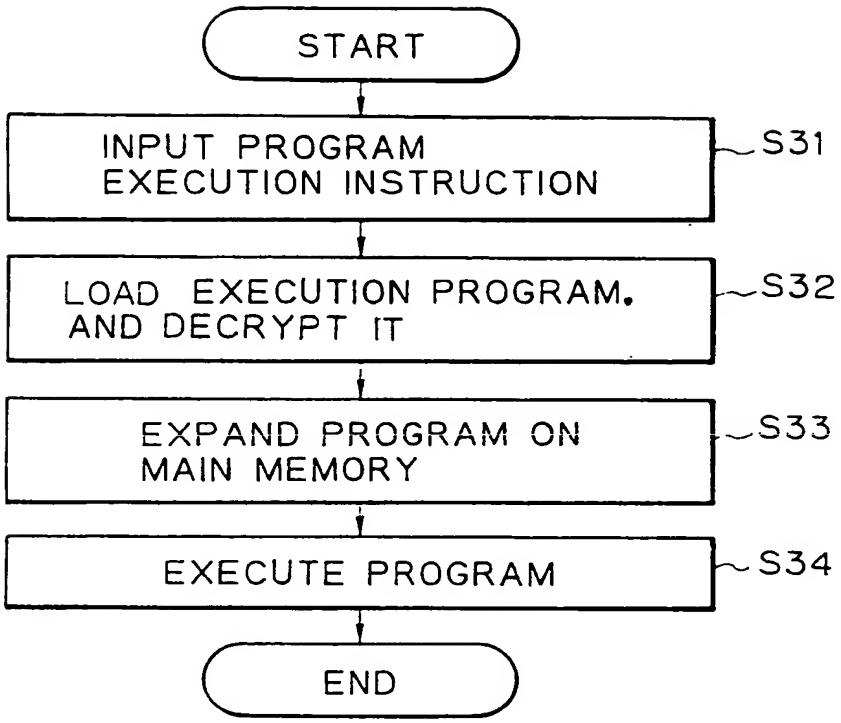

START
INPUT PROGRAM EXECUTION INSTRUCTION
S31
LOAD EXECUTION PROGRAM, AND DECRYPT IT
S32
EXPAND PROGRAM ON MAIN MEMORY
S33
EXECUTE PROGRAM
S34
END

Fig. 10C

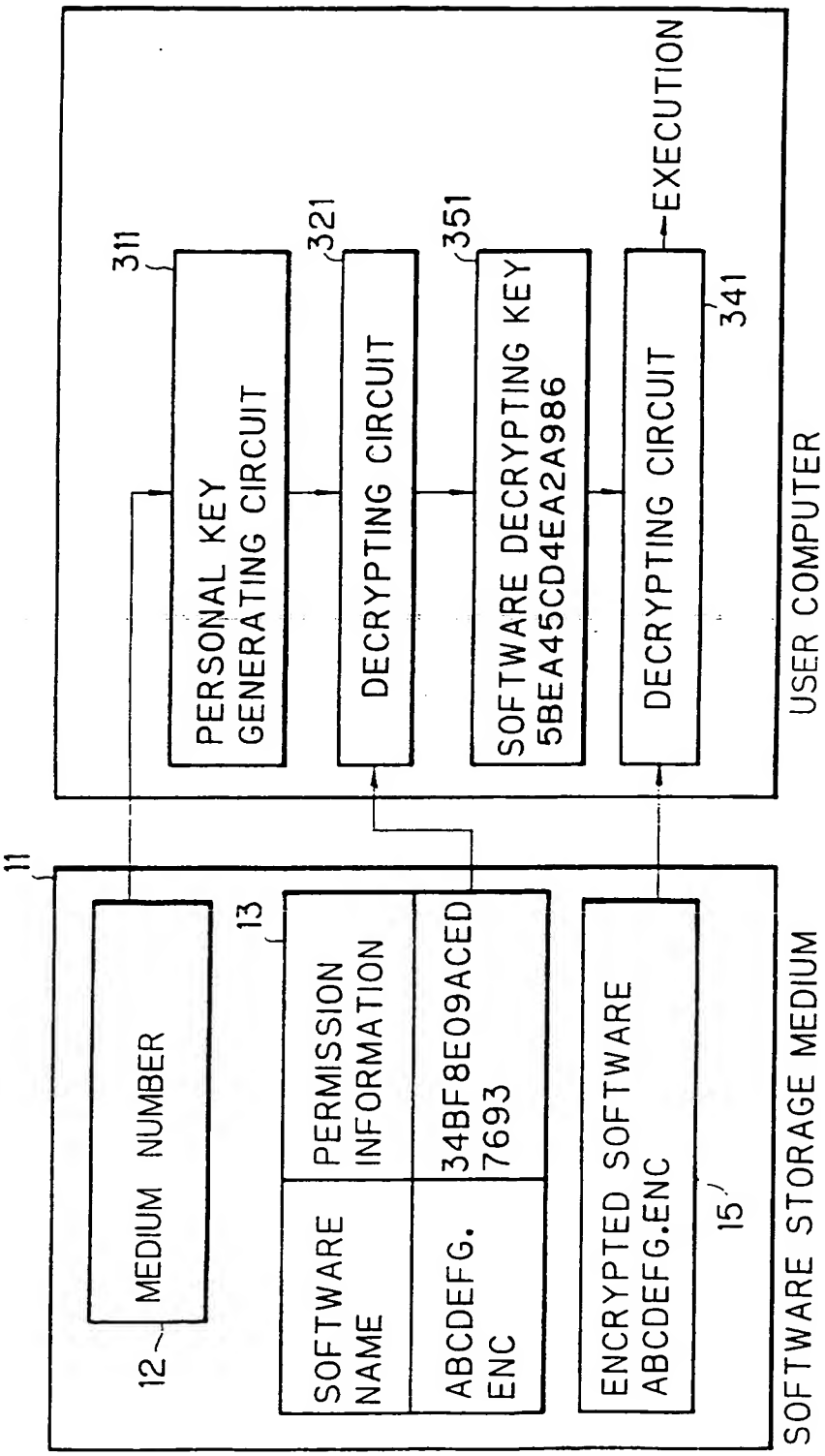

USER COMPUTER
311
PERSONAL KEY GENERATING CIRCUIT
321
DECRYPTING CIRCUIT
351
SOFTWARE DECRYPTING KEY 5BEA45CD4EA2A986
EXECUTION
341
DECRYPTING CIRCUIT
11
SOFTWARE STORAGE MEDIUM
12
MEDIUM NUMBER
13
SOFTWARE NAME
PERMISSION INFORMATION
ABCDEFG. ENC
34BF8E09ACED 7693
15
ENCRYPTED SOFTWARE ABCDEFG.ENC

Fig. 11A

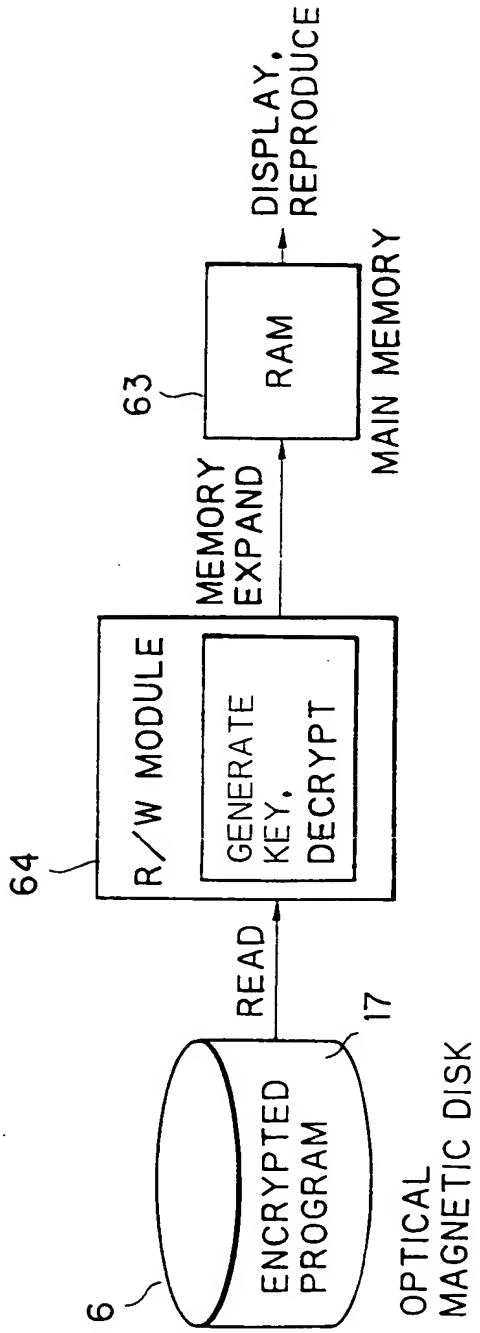

6
ENCRYPTED PROGRAM
OPTICAL MAGNETIC DISK
17
READ
64
R/W MODULE
GENERATE KEY, DECRYPT
MEMORY EXPAND
63
RAM
MAIN MEMORY
DISPLAY, REPRODUCE

Fig. 11B

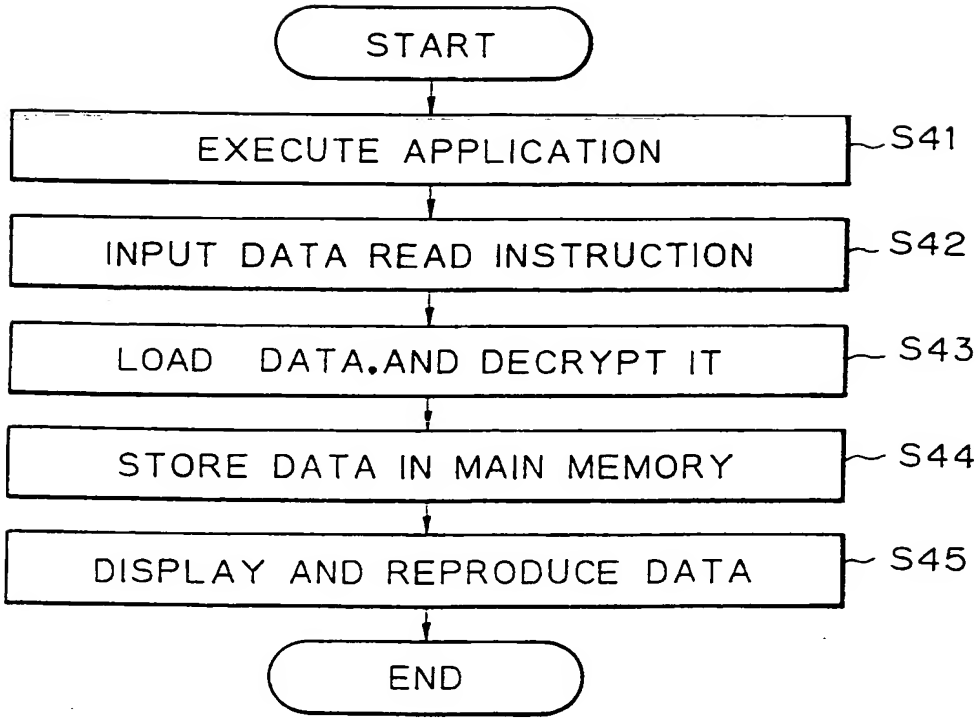

START
EXECUTE APPLICATION
S41
INPUT DATA READ INSTRUCTION
S42
LOAD DATA.AND DECRYPT IT
S43
STORE DATA IN MAIN MEMORY
S44
DISPLAY AND REPRODUCE DATA
S45
END

Fig. 11C

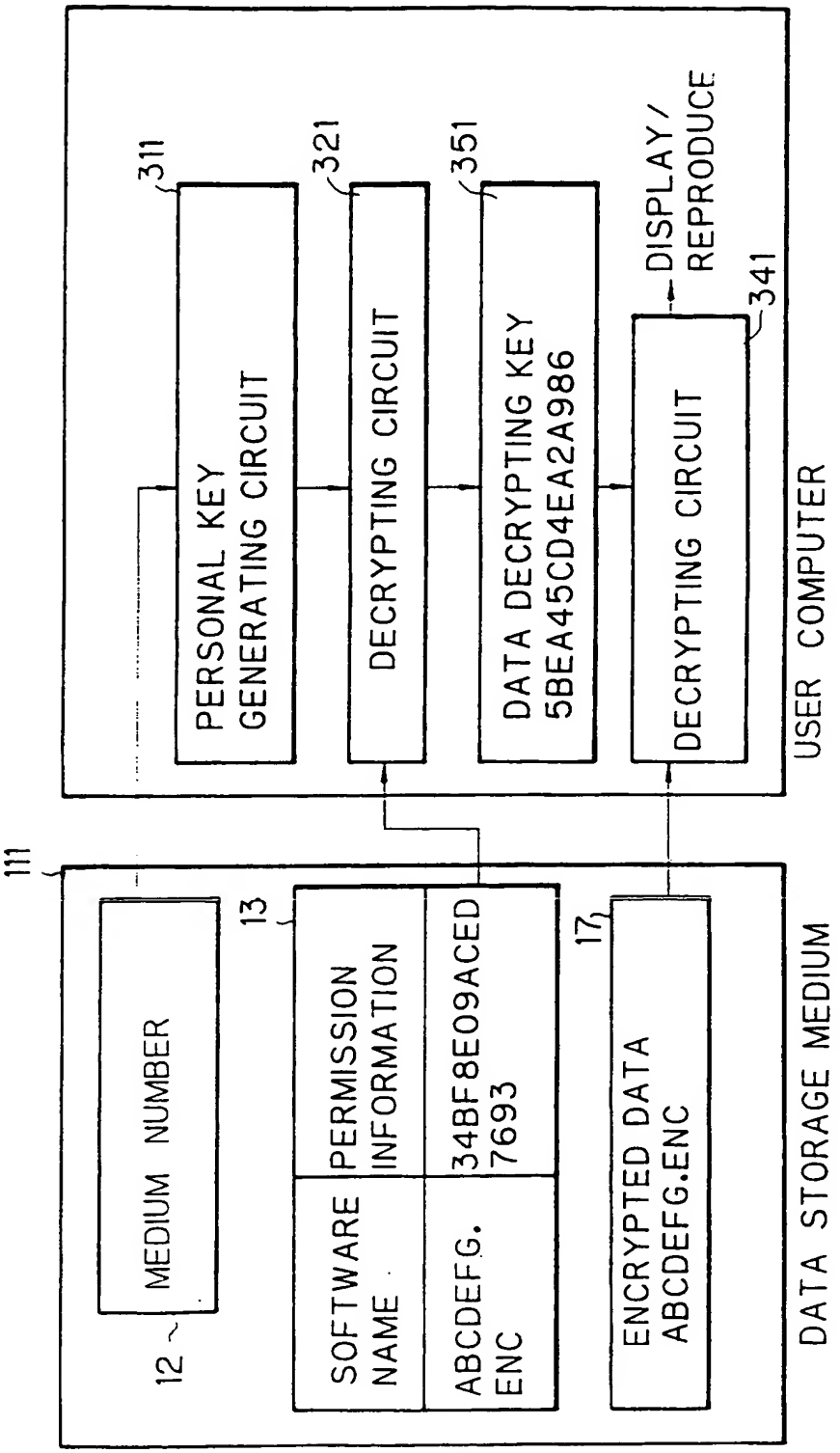

USER COMPUTER
311
PERSONAL KEY GENERATING CIRCUIT
321
DECRYPTING CIRCUIT
351
DATA DECRYPTING KEY 5BEA45CD4EA2A986
DISPLAY/ REPRODUCE
341
DECRYPTING CIRCUIT
111
DATA STORAGE MEDIUM
12
MEDIUM NUMBER
13
PERMISSION INFORMATION
34BF8E09ACED 7693
SOFTWARE NAME
ABCDEFG. ENC
17
ENCRYPTED DATA ABCDEFG.ENC

Fig. 12

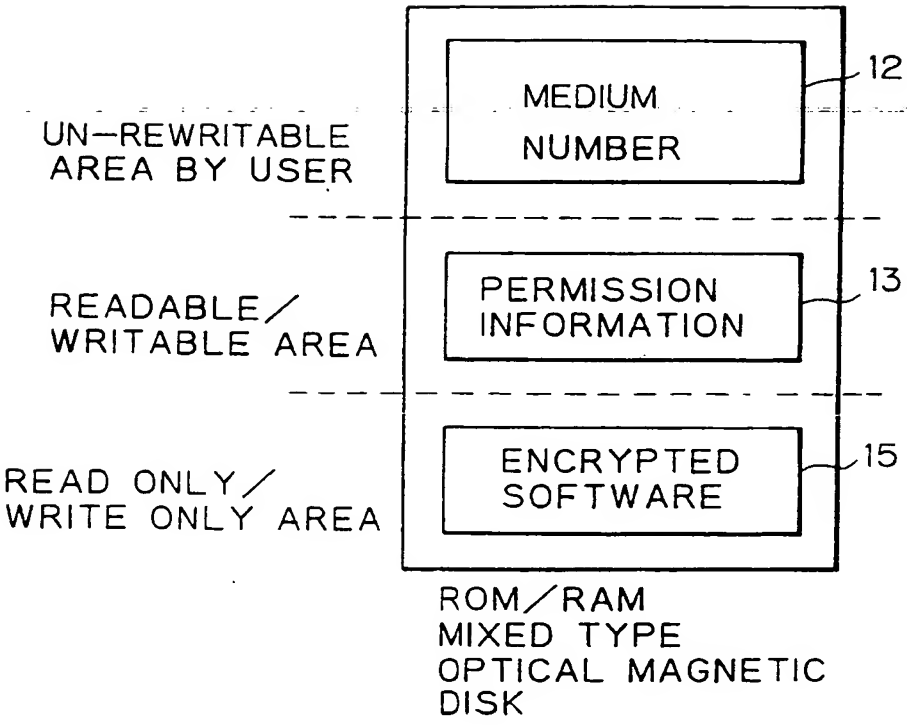

MEDIUM NUMBER
12
UN-REWRITABLE AREA BY USER
PERMISSION INFORMATION
13
READABLE/ WRITABLE AREA
ENCRYPTED SOFTWARE
15
READ ONLY/ WRITE ONLY AREA
ROM/RAM MIXED TYPE OPTICAL MAGNETIC DISK

Fig. 13

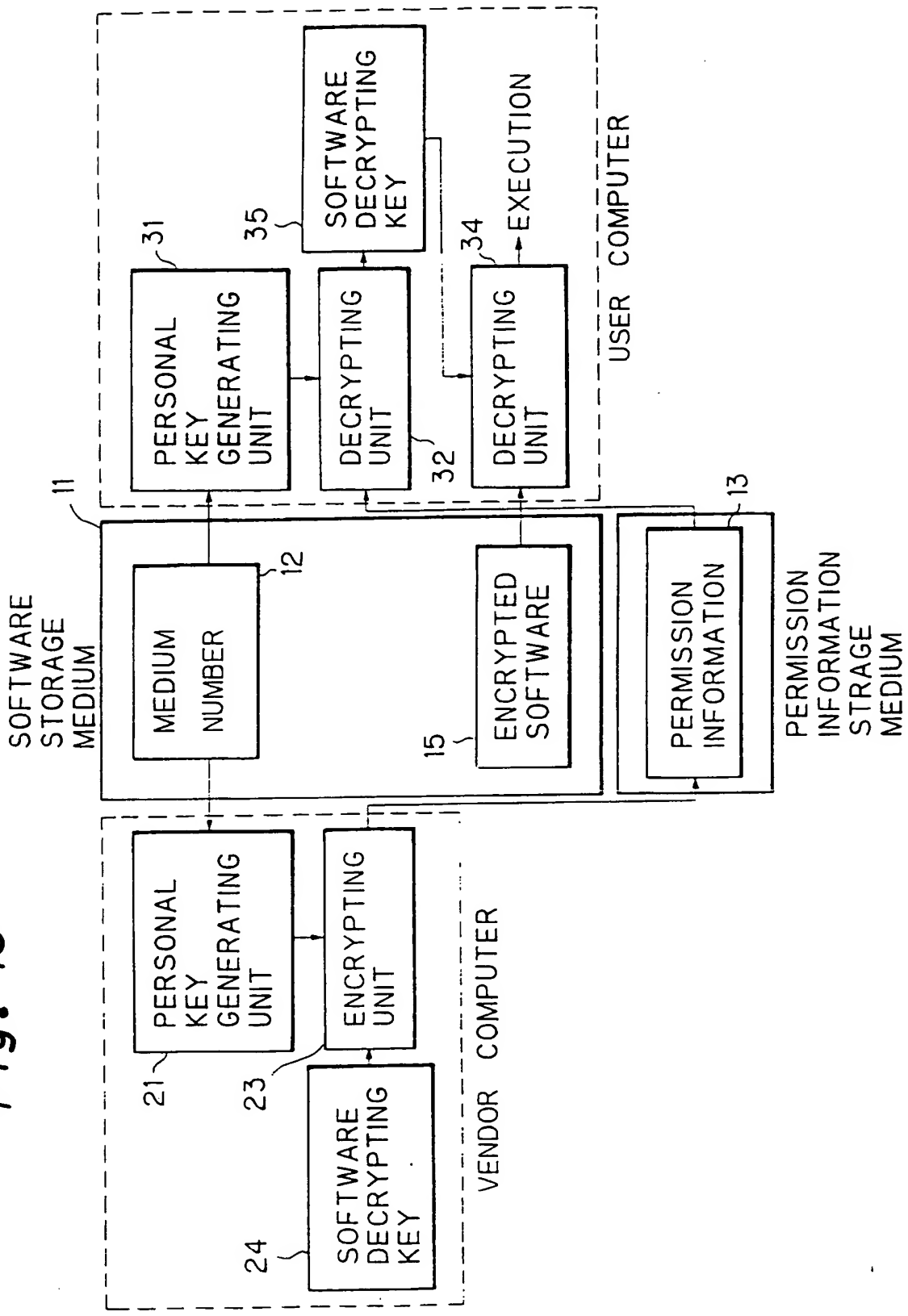
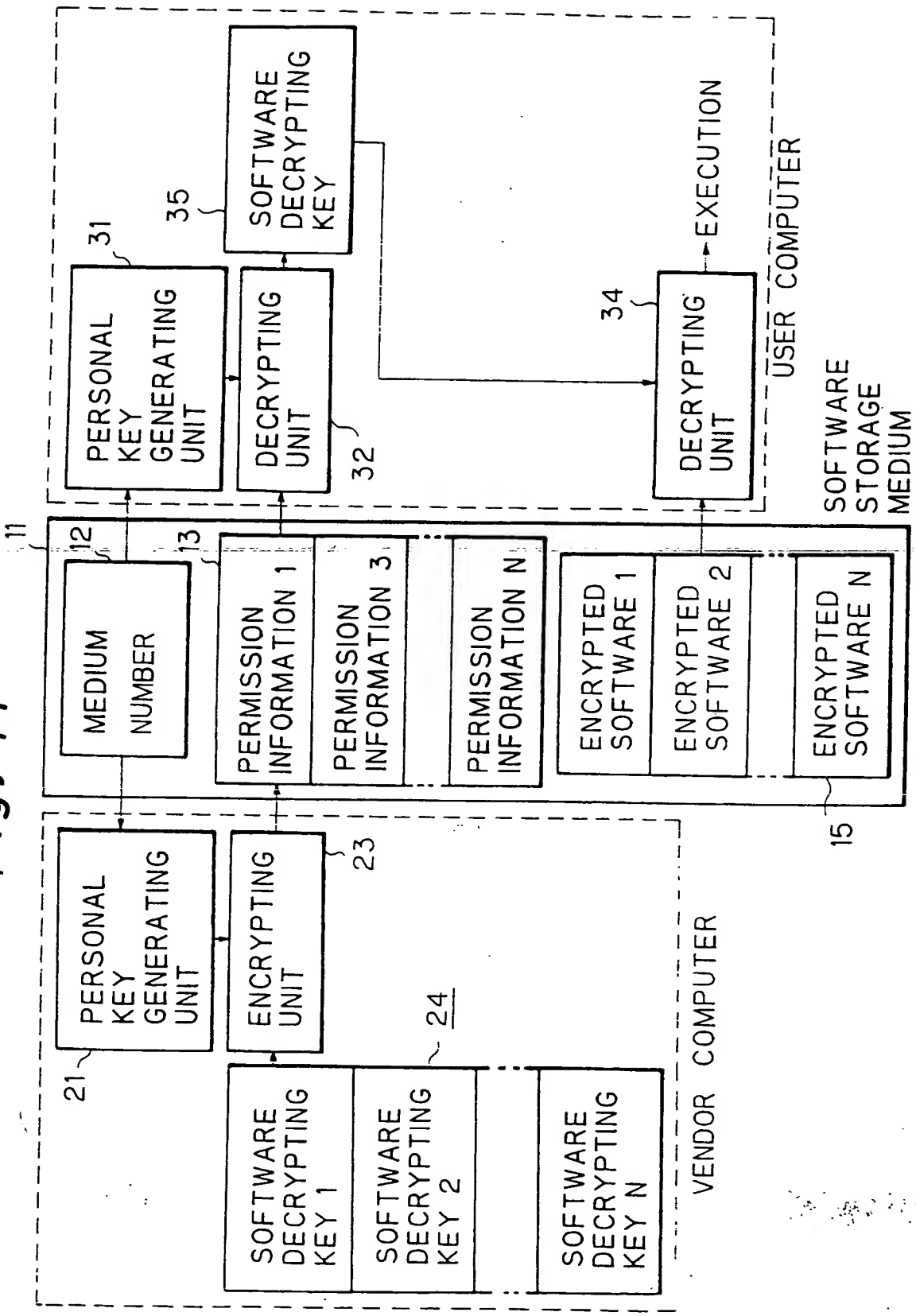

SOFTWARE STORAGE MEDIUM
11
MEDIUM NUMBER
12
PERSONAL KEY GENERATING UNIT
21
ENCRYPTING UNIT
23
SOFTWARE DECRYPTING KEY
24
VENDOR COMPUTER
ENCRYPTED SOFTWARE
15
PERMISSION INFORMATION
13
PERMISSION INFORMATION STRAGE MEDIUM
PERSONAL KEY GENERATING UNIT
31
DECRYPTING UNIT
32
SOFTWARE DECRYPTING KEY
35
DECRYPTING UNIT
34
EXECUTION
USER COMPUTER

Fig. 14


VENDOR COMPUTER
PERSONAL KEY GENERATING UNIT
21
ENCRYPTING UNIT
23
24
SOFTWARE DECRYPTING KEY 1
SOFTWARE DECRYPTING KEY 2
SOFTWARE DECRYPTING KEY N
11
SOFTWARE STORAGE MEDIUM
MEDIUM NUMBER
12
PERMISSION INFORMATION 1
PERMISSION INFORMATION 3
PERMISSION INFORMATION N
13
ENCRYPTED SOFTWARE 1
ENCRYPTED SOFTWARE 2
ENCRYPTED SOFTWARE N
15
USER COMPUTER
PERSONAL KEY GENERATING UNIT
31
SOFTWARE DECRYPTING KEY
35
DECRYPTING UNIT
32
DECRYPTING UNIT
34
EXECUTION